KATEDRA TELEKOMUNIKACJI MULTIMEDIALNEJ I MIKROELEKTRONIKI Wydział Elektroniki i Telekomunikacji Politechnika Poznańska

Rozprawa doktorska

SYNTEZA WIDOKÓW WIRTUALNYCH W RZADKICH SYSTEMACH WIELOKAMEROWYCH DLA ZASTOSOWAŃ W SWOBODNEJ NAWIGACJI

Adrian Dziembowski

Promotor: prof. dr hab. inż. Marek Domański Promotor pomocniczy: dr inż. Olgierd Stankiewicz

Poznań 2018

Spis treści

Spis treści	3
Streszczenie	7
Abstract	9
Spis ważniejszych skrótów i symboli	11
Stosowane definicje	13
1. Wprowadzenie	15
1.1. System swobodnej nawigacji	15
1.2. Synteza widoków wirtualnych	17
1.3. Problemy charakterystyczne dla rzadkich systemów wielokamerowych	21
1.4. Cel i teza pracy	22
1.4.1. Cel rozprawy	22
1.4.2. Teza rozprawy	22
1.5. Przegląd pracy	22
1.6. Publikacje autora rozprawy	24
2. Przegląd stanu wiedzy	29
2.1. Przegląd metod syntezy widoków wirtualnych	29
2.2. Metody syntezy wykorzystujące reprezentację "obraz wielowidokowy + głębia"	30
2.3. Uzupełnianie obrazu syntetycznego	33
2.3.1. Uzupełnianie z uwzględnieniem głębi	35
2.3.2. Uzupełnianie z uwzględnieniem informacji z dalszych widoków rzeczywistych	
2.4. Eliminacja wpływu rozmycia krawędzi w widokach rzeczywistych na jakość syntezy	40
2.5. Korekcja barwna	41
2.6. Podsumowanie	43
3. Metodologia badań	45
3.1. Wprowadzenie	45
3.2. Wybór metody odniesienia	46
3.3. Sposób przeprowadzania eksperymentów	46

Spis treści

3.4. Wykorzystane sekwencje testowe	47
3.5. Inne zbiory materiałów testowych	49
3.6. Pomiar jakości syntezowanych widoków wirtualnych	50
3.6.1. Obiektywny pomiar jakości	50
3.6.2. Subiektywny pomiar jakości	52
4. Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagan	ia 55
4.1. Odsłonięcia	55
4.2. Zalety syntezy z najbliższych widoków rzeczywistych	59
4.2.1. Problem ograniczonej rozdzielczości obrazu	59
4.2.2. Problem nielambertowskich odbić światła	60
4.2.3. Eksperyment – synteza z użyciem różnych widoków rzeczywistych	62
4.3. Zalety syntezy z użyciem informacji z dalszych widoków	65
4.4. Podsumowanie	67
5. Proponowana metoda syntezy widoków wirtualnych	69
5.1. Wprowadzenie	69
5.2. Przetwarzanie pojedynczego widoku rzeczywistego	70
5.3. Łączenie sąsiednich pośrednich widoków wirtualnych	73
5.3.1. Wybór najbliższych widoków rzeczywistych	73
5.3.2. Algorytm łączenia sąsiednich pośrednich widoków wirtualnych	76
5.4. Wypełnianie odsłonięć	78
5.4.1. Wprowadzenie	78
5.4.2. Opracowana technika wypełniania odsłonięć	79
5.4.3. Wyniki eksperymentalne	81
5.5. Uzupelnianie widoku wirtualnego	83
5.6. Podsumowanie	84
6. Proponowana technika filtracji przerzutowanych punktów	87
6.1. Filtracja pośredniej wirtualnej mapy głębi	87
6.1.1. Pęknięcia w wirtualnej mapie głębi	88
6.1.2. Redukcja błędów spowodowanych rozmyciem krawędzi w rzeczywistych mapach gł	ębi 95

6.2. Eliminacja efektów rozmytych krawędzi w widokach rzeczywistych	
6.2.1. Algorytm filtracji	
6.2.2. Przykład działania	
6.3. Wpływ filtracji na jakość syntezy	
6.3.1. Obiektywny pomiar jakości	
6.3.2. Subiektywny pomiar jakości	104
6.4. Podsumowanie	
7. Proponowana technika rzutowania punktów zapewniająca ciągłość obiektów	
7.1. Wprowadzenie	
7.2. Opis zaproponowanej techniki	
7.2.1. Algorytm rzutowania zapewniającego ciągłość obiektów	
7.2.2. Przykład działania	110
7.3. Wyniki eksperymentalne	110
7.3.1. Obiektywny pomiar jakości	113
7.3.2. Subiektywny pomiar jakości	115
8. Proponowana technika korekcji niespójności barwnej widoku wirtualnego	117
8.1. Wprowadzenie	117
8.2. Adaptacyjna korekcja niespójności barwnej	117
8.2.1. Korekcja sąsiednich widoków	119
8.2.2. Korekcja dalszych widoków	123
8.3. Szybka korekcja niespójności barwnej	124
8.4. Wyniki	126
8.4.1. Obiektywny pomiar jakości	
8.4.2. Subiektywny pomiar jakości	130
8.5. Podsumowanie	131
9. Wstępne zwiększanie rozdzielczości widoków rzeczywistych i odpowiadających im m	nap głębi133
9.1. Wprowadzenie	133
9.2. Opis zaproponowanej techniki	135
9.3. Wyniki eksperymentalne	136

Spis treści

10. Oce	10. Ocena efektywności zaproponowanej metody syntezy139					
10.1.	10.1. Wprowadzenie					
10.2.	10.2. Ocena jakości syntezy					
10.	2.1. Jakość mierzona w sposób obiektywny	139				
10.	2.2. Jakość mierzona w sposób subiektywny					
10.	2.3. Jakość syntezy podczas wirtualnej nawigacji	144				
10.3.	Pomiar czasu obliczeń					
11. Synt	11. Synteza czasu rzeczywistego14					
11.1.	11.1. Analiza czasu obliczeń					
11.2. Redukcja czasu obliczeń151						
11.3. Ocena jakości syntezy156						
11.4.	11.4. Podsumowanie					
12. Pod	12. Podsumowanie					
12.1.	12.1. Uzasadnienie tezy rozprawy161					
12.2.	12.2. Przeprowadzone badania					
12.3. Oryginalne osiągnięcia						
13. Bibliografia						
Aneks: Szczegółowe wyniki eksperymentalne						
A1.	Zalety syntezy z najbliższych kamer					
A2.	Wpływ filtracji na jakość syntezy	195				
АЗ.	Rzutowanie zapewniające ciągłość obiektów					
A4.	Korekcja niespójności barwnej					
A5.	Wstępne zwiększanie rozdzielczości					
A6.	Ocena efektywności zaproponowanej metody syntezy					
А7.	Ocena jakości syntezy podczas wirtualnej nawigacji					
A8.	Synteza czasu rzeczywistego	255				

STRESZCZENIE

Rozprawa dotyczy syntezy widoków wirtualnych w systemach swobodnej nawigacji z rzadkim rozmieszczeniem kamer. W początkowej części pracy przedstawiono przegląd znanych z literatury metod syntezy widoków wirtualnych, a także przedyskutowano problemy wynikające z rzadkiego rozmieszczenia kamer.

W rozprawie zaprezentowano opracowaną przez autora metodę syntezy widoków wirtualnych dostosowaną do rzadkich systemów wielokamerowych. Zaproponowana metoda syntezy widoków wirtualnych wykorzystuje następujące opracowane przez autora techniki poprawiające jakość syntezowanych widoków:

- wypełnianie odsłonięć przy użyciu informacji z dalszych widoków rzeczywistych,
- korekcja niespójności barwnej,
- filtracja rozmytych krawędzi w widokach rzeczywistych wykorzystująca analizę gradientów,
- rzutowanie punktów do widoku wirtualnego zapewniające ciągłość obiektów,
- wstępne zwiększanie rozdzielczości rzeczywistych widoków i map głębi.

Ponadto wykazano, iż możliwe jest stworzenie metody syntezy widoków rzeczywistych wykorzystującej reprezentację "obraz wielowidokowy + glębia" umożliwiającej syntezę dobrej jakości widoków wirtualnych w czasie rzeczywistym. Tym samym jest możliwe stworzenie systemu swobodnej nawigacji, w którym widz może się wirtualnie przemieszczać dookoła sceny zarejestrowanej przy pomocy rzadkiego systemu wielokamerowego.

Streszczenie

ABSTRACT

The dissertation is related to the virtual view synthesis for sparse multicamera systems. At the beginning existing virtual view synthesis methods are described. Moreover, problems caused by sparse camera arrangement are discussed.

In the dissertation, the author's virtual view synthesis method suitable for sparse multicamera systems is presented. Proposed virtual view synthesis method uses original techniques improving the quality of synthesized virtual views:

- disocclusion filling using information from non-neighboring real views,
- color inconsistence correction,
- gradient-based filtering of blurred edges in real views,
- technique of point projection preserving objects' continuity,
- technique of preliminary increasing resolution of the real views and depth maps.

Moreover, it was proved, that it is possible to create MVD-based virtual view synthesis method, which allows to synthesize good-quality virtual views in the real time. Therefore it is possible to create a free-navigation system, where a viewer may virtually navigate around the scene acquired by the sparse multicamera system.

Abstract

Spis ważniejszych skrótów i symboli

K	_	macierz parametrów wewnętrznych kamery
Р	_	macierz projekcji kamery
RT	_	macierz parametrów zewnętrznych kamery
BBB	_	Big Buck Bunny,
		grupa syntetycznych sekwencji testowych
DIBR	_	Depth Image Based Rendering,
		synteza wykorzystująca reprezentację MVD
FIV	_	Free-Viewpoint Television, telewizia swobodnego punktu widzenia
ITU-R	_	International Telecommunication Union (Radiocommunication Sector), Międzynarodowy Związek Telekomunikacyjny (Sektor Radiokomunikacji)
ITU-T	_	International Telecommunication Union (Telecommunication Standarization Sector), Międzynarodowy Związek Telekomunikacyjny (Sektor Normalizacji Telekomunikacji)
MPEG	_	Moving Picture Experts Group, grupa robocza ISO/IEC zajmująca się rozwojem kodowania dźwięku i obrazu
MVD	-	Multiview Video plus Depth, "obraz wielowidokowy + glębia": sposób reprezentacji sekwencji wielowidokowych
MVS	_	MultiView Synthesis, zaproponowana w rozprawie metoda syntezy widoków wirtualnych
РС	_	Pair Comparison, metoda subiektywnego badania jakości
PSNR	-	Peak Signal to Noise Ratio, miara jakości obrazu bazująca na stosunku sygnału do szumu
rtMVS	_	realtime MultiView Synthesis,
		zaproponowana w rozprawie metoda syntezy widoków w czasie rzeczywistym
SSIM	_	Structural Similarity,
		miara podobieństwa strukturalnego
VSRS	_	View Synthesis Reference Software,
		oprogramowanie odniesienia grupy MPEG do syntezy widoków wirtualnych
YCbCr	_	przestrzeń barw (luminancja Y oraz 2 chrominancje: Cb i Cr)

Spis ważniejszych skrótów i symboli

STOSOWANE DEFINICJE

Rzadki system wielokamerowy – system wielokamerowy z rzadkim rozmieszczeniem kamer; charakteryzujący się dużym kątem między osiami optycznymi poszczególnych kamer lub dużymi odległościami między kamerami.

Kamera rzeczywista – kamera, którą zarejestrowano jeden widok wielowidokowej sekwencji wizyjnej; fizyczne urządzenie rejestrujące obraz.

Widok rzeczywisty - obraz zarejestrowany przez rzeczywistą kamerę.

Rzeczywista mapa głębi – mapa głębi, której punkty odpowiadają punktom widoku rzeczywistego.

Kamera wirtualna – nieistniejące fizycznie, rozważane teoretycznie urządzenie, z którego obraz jest generowany podczas syntezy; przyjmuje się, iż ma ona określoną charakterystykę optyczną i określone położenie w przestrzeni.

Widok wirtualny – obraz generowany podczas syntezy, odpowiadający widokowi rejestrowanemu przez kamerę wirtualną.

Pośredni widok wirtualny – widok wirtualny zawierający punkty przerzutowane wyłącznie z jednego widoku rzeczywistego.

Wirtualna mapa głębi – mapa głębi, której punkty odpowiadają punktom widoku wirtualnego.

Pośrednia wirtualna mapa glębi – mapa glębi, której punkty odpowiadają punktom pośredniego widoku wirtualnego.

Odsłonięcie – ang. *disocclusion*, obszar widoku wirtualnego zawierający obiekty niewidoczne (przysłonięte) w pewnej grupie widoków rzeczywistych.

Uzupełnianie – ang. *inpainting* (czasem: *hole-filling*), wypełnianie niezsyntezowanych obszarów widoku wirtualnego przy użyciu informacji z otoczenia (lub innych ramek sekwencji).

Dalszy widok rzeczywisty – obraz zarejestrowany przez dalszą kamerę rzeczywistą, tj. kamerę niesąsiadującą z rozważaną kamerą wirtualną.

Stosowane definicje

1. WPROWADZENIE

1.1. System swobodnej nawigacji

Tradycyjna telewizja umożliwia widzowi oglądanie zarejestrowanej sceny tylko z jednego punktu widzenia, z góry wybranego przez realizatora bądź reżysera. W przypadku tradycyjnej telewizji widz nie ma możliwości zmiany punktu widzenia.

Krokiem naprzód w kierunku zapewnienia widzowi wrażenia znajdowania się w scenie są monitory autostereoskopowe [Lee'13A, Urey'11]. W przypadku tego typu monitorów zastosowanie większej liczby widoków (np. 10 – 30), które są obserwowane w zależności od kąta spoglądania na monitor, pozwala odzwierciedlić efekt paralaksy. Dzięki temu użytkownik jest w stanie oglądać obraz przestrzenny bez konieczności stosowania okularów. Niestety, pośród innych problemów, jak choćby zbyt mała rozdzielczość matryc czy też cena takich monitorów, oglądanie obrazu stereoskopowego może być męczące dla wzroku [Howarth'11]. Sposób odzwierciedlania głębi sceny na plaskim monitorze powoduje konflikt dwuocznych i jednoocznych wskazówek glębi, a mianowicie zbieżności i akomodacji wzroku [Domański'11]. Ponadto, monitory autostereoskopowe, pomimo umożliwienia widzowi obserwacji zarejestrowanej sceny pod różnymi kątami, posiadają podstawowe ograniczenie, jakim jest niewielki zakres zmiany punktu widzenia – zakres ten bowiem ma odpowiadać ruchom głowy widza znajdującego się przed monitorem.

Wyżej wymienione ograniczenia nie istnieją w przypadku telewizji swobodnego punktu widzenia (FTV, *Free-Viewpoint Television*) – rodzaju telewizji interaktywnej [Fehn'06, Palacio'17, Warmiński'04], w której użytkownik ma możliwość **swobodnej nawigacji** w zarejestrowanej scenie. W takim systemie widz sam może ustalać swój punkt widzenia i kierunek patrzenia [Domański'14D, Lee'15, Tanimoto'02, Tanimoto'10, Tanimoto'12], nie ograniczając się wyłącznie do niewielkiego przesuwania się na przykład w lewą bądź prawą stronę.

Jedną z metod umożliwiających użytkownikowi zmianę punktu widzenia jest akwizycja sceny za pomocą systemu złożonego z wielu kamer – od kilku [Dziembowski'14B, Goorts'12, Zitnick'04], przez kilkadziesiąt [Domański'18B, Stankiewicz'18] do nawet kilkuset [Fujii'06, Tanimoto'09B]. Najprostszym ustawieniem kamer jest ustawienie liniowe [Domański'09B, Feldmann'08, Goorts'12, Ho'08, Tanimoto'08B, Um'08], w którym osie optyczne wszystkich kamer są równoległe. Takie ustawienie może być jednak niewystarczające dla potrzeb swobodnej nawigacji [Kawamura'13]. Bardziej zaawansowane systemy charakteryzują się ustawieniem kamer w sposób inny, niż liniowy. Drugim często stosowanym sposobem ustawienia kamer jest ich równomierne rozmieszczenie na łuku [Domański'15A, Domański'15C, Domański'15E, Goorts'14A, Kovacs'15A, Kovacs'15B, Zitnick'04]. Równomierne rozmieszczenie kamer na łuku niekoniecznie jest rozmieszczeniem najlepszym (jak wykazano w [Domański'16A, Domański'16G, Stankiewicz'18]),

dlatego też stworzono system, w którym kamery ustawione na łuku grupowane są parami [Domański'16B, Domański'16F, Domański'17]. Dla potrzeb transmisji sportowych, kamery można umieścić wokół hali sportowej lub stadionu z wykorzystaniem istniejącej architektury, a więc bezpośrednio na ścianach hali czy trybunach stadionu: równomiernie [Goorts'12, Goorts'14A] lub w parach [Domański'16B, Domański'18B, Stankiewicz'18]. W ogólności, w praktycznych systemach ustawienie kamer może być bardzo nieregularne.

System telewizji swobodnego punktu widzenia może znaleźć wiele zastosowań, jak chociażby wymienione w pracy [Domański'17] wydarzenia kulturalne (np. spektakle teatralne), kursy i instruktaże, na przykład kursy tańca, naprawy samochodu czy też kurs pierwszej pomocy. Przede wszystkim jednak, największy potencjał wydaje się mieć transmisja wydarzeń sportowych takich jak piłka nożna [Goorts'12, Goorts'14A, Kawamura'13, Tanimoto'13], zespołowe sporty halowe (np. siatkówka lub koszykówka) [Domański'18B] bądź też sporty indywidualne jak szermierka czy judo [Domański'17, Wegner'17B].

Podstawowym założeniem systemu telewizji swobodnego punktu widzenia jest umożliwienie widzowi swobodnej nawigacji w scenie – czy to za pomocą dedykowanego odbiornika [Bondarev'11], czy też komputera bądź telefonu komórkowego [Domański'17, Stankiewicz'18]. Swobodne przemieszczanie się w scenie wymaga jednak płynnego wirtualnego ruchu widza, a więc pula punktów widzenia, jaką on dysponuje nie może być ograniczona wyłącznie do widoków zarejestrowanych przez kamery systemu wielokamerowego. Widz musi mieć możliwość oglądania sceny również z innych, dowolnie wybranych pozycji. Ideę systemu swobodnej nawigacji przedstawiono na rysunku 1.1. Co należy podkreślić, ideą systemu telewizji swobodnego punktu widzenia jest swobodna nawigacja użytkownika w naturalnej, zarejestrowanej przez wiele kamer scenie.

W pracy skupiono się na syntezie widoków wirtualnych dla swobodnej nawigacji. Niemniej jednak, synteza widoków wirtualnych ma więcej zastosowań, niż tylko zapewnianie swobodnej nawigacji widza. Jako przykład podać można kompresję sekwencji wielowidokowych [Domański'13, Domański'15B, Domański'15F, Mueller'13, Stankowski'15, Zou'14], tworzenie brakujących widoków dla monitora autostereoskopowego [Tanimoto'12] i możliwość wirtualnej zmiany bazy systemu dwukamerowego, którym zarejestrowano sekwencję stereoskopową, co może znaleźć zastosowanie w przypadku chęci wyświetlenia tej samej sekwencji na dużym i małym ekranie stereoskopowym – na przykład w kinie i na ekranie telefonu komórkowego [Held'08].



Rysunek 1.1. Ilustracja idei swobodnej nawigacji; kolorem czarnym oznaczono rzeczywiste kamery systemu wielokamerowego, kolorem pomarańczowym – kamerę wirtualną, z której użytkownik systemu ogląda obraz i której pozycję może on w dowolny sposób zmieniać

1.2. SYNTEZA WIDOKÓW WIRTUALNYCH

W celu umożliwienia widzowi płynnego wirtualnego przemieszczania się w zarejestrowanej scenie, pośrednie widoki (widoki wirtualne) muszą zostać zsyntezowane, a więc stworzone na podstawie widoków rzeczywistych, czyli widoków zarejestrowanych przez rzeczywiste kamery systemu wielokamerowego. Co więcej, oczekuje się, że różnica jakości pomiędzy widokami rzeczywistymi i wirtualnymi będzie jak najmniejsza, aby widz nie był w stanie stwierdzić, czy ogląda obrazy zarejestrowane przez kamery, czy też obrazy syntetyczne [Lafruit'16].

Do przeprowadzenia syntezy widoków wirtualnych wymagane są widoki rzeczywiste oraz informacja przestrzenna o scenie (informacja o rozmieszczeniu obiektów w scenie) i samym systemie wielokamerowym (informacja o rozmieszczeniu kamer). Kamery wchodzące w skład systemu wielokamerowego mogą być opisane poprzez parametry wewnętrzne i zewnętrzne [Cyganek'02, Heyden'05], najczęściej opisane w postaci macierzowej.

Macierz **parametrów wewnętrznych K** opisuje pojedynczą kamerę, a przedstawić ją można w sposób następujący [Domański'11]:

$$\mathbf{K} = \begin{bmatrix} f \cdot s_x & f \cdot s_k & o_x \\ 0 & f \cdot s_y & o_y \\ 0 & 0 & 1 \end{bmatrix},\tag{1.1}$$

gdzie f jest ogniskową kamery, s_x i s_y okresem próbkowania w kierunku poziomym i pionowym, s_k skośnością przetwornika kamery (zazwyczaj $s_k \approx 0$), a o_x i o_y położeniem punktu zasadniczego kamery (wyrażonym w okresach próbkowania).

Macierz **parametrów zewnętrznych** opisuje rotację (obrót) kamery, a także jej położenie względem początku układu współrzędnych wspólnego dla wszystkich kamer systemu:

$$\mathbf{RT} = [\mathbf{R}_{3\times3} | \mathbf{T}_{3\times1}], \tag{1.2}$$

gdzie **R** jest macierzą rotacji (opisującą obrót kamery wokół trzech osi wspólnego układu współrzędnych, a **T** wektorem translacji (opisującym przesunięcie kamery od środka wspólnego układu współrzędnych). Macierz **K** jest macierzą kwadratową o 3 kolumnach i 3 wierszach, a macierz **T** jest 3-elementowym wektorem kolumnowym.

Oprócz informacji o przestrzennym rozmieszczeniu kamer synteza widoków wirtualnych wymaga również informacji o rozmieszczeniu obiektów w rejestrowanej scenie. Informacja ta może być reprezentowana na różne sposoby, jak choćby poprzez chmurę punktów [Wei'13, Wu'17] czy przestrzeń promieni [Tanimoto'02, Tanimoto'06, Dziembowski'14A, Kim'13], jednakże najpopularniejszą i najczęściej stosowaną reprezentacją jest reprezentacja "obraz wielowidokowy + glębia" (MVD, *Multiview Video plus Depth*), gdzie informacja przestrzenna zawarta jest w mapach glębi [Fehn'03, Mueller'11, Zinger'10].

Znając parametry kamer i dokładne położenie obiektów zarejestrowanych przez kamery, można zsyntezować widok wirtualny poprzez rzutowanie punktów z widoków rzeczywistych. Każdy punkt rzeczywistego widoku rzutowany jest zgodnie ze wzorem:

- ----

$$\begin{bmatrix} x'\\ y'\\ z'\\ w' \end{bmatrix} = \mathbf{P}_{\mathbf{V}} \cdot \mathbf{P}_{\mathbf{R}}^{-1} \cdot \begin{bmatrix} x_{R}\\ y_{R}\\ z_{R}\\ 1 \end{bmatrix}, \qquad (1.3)$$

gdzie: x_R , y_R są współrzędnymi danego punktu w widoku rzeczywistym, z_R odpowiadającą mu głębią (odległością w przestrzeni trójwymiarowej pomiędzy punktem a płaszczyzną przetwornika kamery rzeczywistej), \mathbf{P}_R i \mathbf{P}_V są macierzami projekcji (patrz: (1.5)), odpowiednio dla kamery rzeczywistej i wirtualnej, a pozycja i głębia odpowiadającego punktu w widoku wirtualnym (oznaczane odpowiednio x_V , y_V i z_V) wyznaczane są jako:

$$x_V = \frac{x'}{w'}; \qquad y_V = \frac{y'}{w'}; \qquad z_V = \frac{z'}{w'}.$$
 (1.4)

Macierze projekcji wyznaczane są ze wzoru:

$$\mathbf{P} = \begin{bmatrix} \mathbf{K}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{R}_{3\times3} & \mathbf{T}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix},$$
(1.5)

gdzie **K**, **R** i **T** są odpowiednio macierzą parametrów wewnętrznych (1.1), macierzą rotacji i wektorem translacji. W indeksach dolnych umieszczono rozmiary poszczególnych macierzy.

Rzutowanie pojedynczego punktu widoku rzeczywistego do widoku wirtualnego przedstawiono na rysunku 1.2. Współrzędne punktu w widoku rzeczywistym (x_R, y_R) wraz z odpowiadającą mu głębią z_R są mnożone przez odwrotną macierz projekcji kamery rzeczywistej \mathbf{P}_R^{-1} w celu wyznaczenia pozycji punktu w przestrzeni trójwymiarowej (X, Y, Z). Następnie współrzędne te są mnożone przez macierz projekcji kamery wirtualnej \mathbf{P}_V , czego wynikiem jest pozycja (x_V, y_V) i głębia z_V odpowiadającego punktu w widoku wirtualnym.



Rysunek 1.2. Rzutowanie punktu (x_R, y_R) widoku rzeczywistego o glębi z_R do widoku wirtualnego; odpowiadający punkt widoku wirtualnego znajduje się w położeniu (x_V, y_V) i ma glębię z_V

Zauważyć należy, iż wyznaczona w drodze rzutowania pozycja punktu w widoku wirtualnym opisana jest przez parę liczb wymiernych (x_V, y_V) . Widok wirtualny jest obrazem o rozdzielczości takiej, jak rozdzielczość widoku rzeczywistego, a więc pozycja dowolnego punktu w widoku wirtualnym musi być opisana parą liczb całkowitych. Z tego powodu, współrzędne punktu są zaokrąglane do najbliższych wartości całkowitych:

$$(\widetilde{x_V}, \widetilde{y_V}) = ([x_V], [y_V]), \qquad (1.6)$$

gdzie operator [·] oznacza zaokrąglenie do najbliższej liczby całkowitej.

Konieczność zaokrąglania pozycji przerzutowanego punktu wiąże się ze stratą dokładności rzutowania, a tym samym przekłada się na jakość syntezowanego widoku. Problem zaprezentowano na rysunku 1.3.



Rysunek 1.3. Problem ograniczonej rozdzielczości widoku wirtualnego; współrzędne przerzutowanego punktu (x_V, y_V) muszą zostać zaokrąglone do liczb całkowitych $(\widetilde{x_V}, \widetilde{y_V})$: (6, 16)

W dalszych rozważaniach często zachodzi konieczność stwierdzenia, które punkty widoku wirtualnego odpowiadają punktom z widoków rzeczywistych (i odwrotnie). Odpowiedniość taka najczęściej nie występuje w sensie matematycznym, dlatego w rozprawie posłużono się odpowiedniością z dokładnością do współrzędnych całkowitych.

W rozprawie przyjęto, iż punkt $m_A = (x_A, y_A)$ w widoku wirtualnym będzie nazywany punktem przerzutowanym z widoku rzeczywistego, jeżeli:

$$\bigvee_{m_R} [x_V] = x_A \wedge [y_V] = y_A , \qquad (1.7)$$

gdzie:

- operator [·] oznacza zaokrąglenie do najbliższej liczby całkowitej,
- $-m_R$ jest dowolnym punktem widoku rzeczywistego,
- x_V i y_V określają pozycję w widoku wirtualnym punktu m_R przerzutowanego z widoku rzeczywistego zgodnie z (1.3).

Analogicznie, jeżeli:

$$\bigwedge_{m_R} [x_V] \neq x_A \lor [y_V] \neq y_A , \qquad (1.8)$$

punkt $m_A = (x_A, y_A)$ będzie określany jako **punkt nieprzerzutowany z widoku rzeczywistego**.

1.3. PROBLEMY CHARAKTERYSTYCZNE DLA RZADKICH SYSTEMÓW WIELOKAMEROWYCH

Synteza widoków wirtualnych w rzadkich systemach wielokamerowych jest zdecydowanie trudniejsza niż w przypadku systemów gęstych. Zwiększona trudność spowodowana jest dużą odległością pomiędzy poszczególnymi kamerami (wynoszącą np. metr [Goorts'14B], półtora metra [Domański'18B] czy 10 metrów [Goorts'14C]) lub dużym kątem pomiędzy ich osiami optycznymi (wynoszącym np. 10° [Domański'14C], 15° [Domański'16F, Goorts'14C], a w przypadku kamer umieszczonych na dwóch prostopadłych ścianach nawet 90° [Domański'18B]). Skutkuje to trzema podstawowymi problemami charakterystycznymi dla rzadkich systemów wielokamerowych:

- 1. Mniejszy obszar sceny jest widoczny jednocześnie w różnych widokach rzeczywistych, co skutkuje powstawaniem odsłonięć w widoku wirtualnym.
- 2. Ograniczona rozdzielczość obrazu skutkująca powstawaniem pęknięć (ang. *cracks*) nieciągłości obiektów w widoku wirtualnym.
- 3. Nielambertowskie odbicia światła skutkujące różną barwą tych samych obiektów w różnych widokach rzeczywistych.

Niezależnie od charakterystyki systemu wielokamerowego, w syntezowanych widokach wirtualnych występują odsłonięcia [Buyssens'15], czyli obszary niewidoczne w widokach rzeczywistych użytych do syntezy. Jak pokazano jednak w rozdziale 4.1, w przypadku sekwencji zarejestrowanych przy użyciu rzadkich systemów wielokamerowych powierzchnia takich obszarów jest zdecydowanie większa. W takim przypadku zastosowanie znanych metod uzupełniania (ang. *inpainting*) obrazu (np. [Buyssens'17, Huang'14, LeMeur'13, Luo'17, Plath'13, Zhu'16A], rozdział 2.3) nie pozwala na osiągnięcie zadowalającej jakości widoków wirtualnych [Fachada'18A]. Powierzchnię obszarów odsłoniętych można zmniejszyć używając do syntezy większą liczbę widoków rzeczywistych (np. [Ceulemans'18, Fachada'18A, Jun-Te'15, Li'18]).

W przypadku rzadkich systemów wielokamerowych należy jednak wziąć pod uwagę dwa pozostałe problemy, a więc ograniczoną rozdzielczość obrazu (rozdział 4.2.1) i nielambertowskie odbicia światła (rozdział 4.2.2). Oba problemy skutkują niepoprawnym rzutowaniem punktów z dalszych widoków rzeczywistych. Z powodu ograniczonej rozdzielczości obrazu nie jest dokładnie wyznaczana pozycja przerzutowanych punktów, a z powodu nielambertowskich odbić światła rzutowana jest niepoprawna barwa tychże punktów.

Rozwiązanie trzech wymienionych problemów jest kluczowe dla opracowania metody syntezy widoków wirtualnych dostosowanej do celów swobodnej nawigacji w rzadkich systemach wielokamerowych.

1.4. CEL I TEZA PRACY

1.4.1. Cel rozprawy

Celem pracy jest rozwiązanie problemów utrudniających syntezę dobrej jakości widoków wirtualnych w praktycznych systemach telewizji swobodnego punktu widzenia i opracowanie efektywnej metody syntezy widoków wirtualnych dla celów swobodnej nawigacji. Niniejsza rozprawa ma więc na celu poszerzenie stanu wiedzy w tej dziedzinie i umożliwienie rozwoju tego rodzaju systemów. W rozprawie zaprezentowano nową, wydajną metodę syntezy widoków wirtualnych dostosowaną do rzadkiego rozmieszczenia kamer.

1.4.2. TEZA ROZPRAWY

Możliwe jest opracowanie wydajnej metody syntezy widoków wirtualnych dobrej jakości poprzez wykorzystanie dalszych widoków rzeczywistych, filtrację krawędzi w widokach rzeczywistych i odpowiadających im mapach glębi oraz korekcję niespójności barwnej widoku wirtualnego.

1.5. PRZEGLĄD PRACY

W dalszej części pracy opisano kolejne zagadnienia dotyczące syntezy widoków wirtualnych ze szczególnym uwzględnieniem rzadkich systemów wielokamerowych.

W rozdziale 2. dokonano przeglądu istniejących rozwiązań w zakresie syntezy widoków wirtualnych. Opisano szereg metod syntezy bazujących na różnych założeniach i reprezentacjach sygnału wizyjnego.

W rozdziale 3. opisano wykorzystaną w rozprawie metodologię badań, sposób przeprowadzania eksperymentów, zbiór sekwencji testowych wraz z uzasadnieniem ich wykorzystania, a także dobór metod estymacji jakości syntezowanych widoków wirtualnych.

W rozdziale 4. dokonano analizy charakterystyki systemów wielokamerowych z rzadkim rozmieszczeniem kamer, skupiając się na aspektach zmniejszających jakość syntezowanych widoków, a więc różnej charakterystyce barwnej poszczególnych widoków rzeczywistych oraz dużej powierzchni odsłoniętych obszarów w widoku syntezowanym. W rozdziale pokazano wady i zalety syntezy widoków wirtualnych używającej informacji z większej liczby widoków rzeczywistych.

W rozdziałach 5 – 11 przedstawiono szczegółowo oryginalne osiągnięcia rozprawy wraz z wynikami eksperymentalnymi.

Rozdział 5. zawiera szczegółowy opis opracowanej przez autora rozprawy wielowidokowej metody syntezy widoków wirtualnych (MVS – *MultiView Synthesis*) wraz z krótką charakterystyką dodatkowych zaproponowanych technik zwiększających jakość syntezowanych widoków, tj. filtracji

widoku wirtualnego i odpowiadającej mu mapy glębi, algorytmu rzutowania wymuszającego ciągłość obiektów, korekcji niespójności barwnej oraz wstępnego zwiększania rozdzielczości wejściowych widoków i map glębi.

W rozdziałach 6 – 9 dokonano szczególowej analizy poszczególnych zaproponowanych przez autora rozprawy technik poprawiających jakość syntezy.

W rozdziale 6. opisano sposób filtracji wirtualnej mapy glębi w celu usunięcia blędów wynikających z niepoprawnej informacji o glębi niektórych punktów widoków rzeczywistych oraz nieciągłości będących efektem ograniczonej rozdzielczości obrazu i glębi. Ponadto, opisano sposób usuwania pojawiających się w widoku wirtualnym nieistniejących krawędzi wynikających z rozmycia krawędzi obiektów w widokach rzeczywistych oraz wzajemnego przesunięcia krawędzi w widokach rzeczywistych i odpowiadających im mapach glębi.

W rozdziale 7. opisano sposób ulepszonego rzutowania informacji z widoków rzeczywistych do widoku wirtualnego. W zaproponowanym podejściu informacja nie jest rzutowana niezależnie dla wszystkich punktów obrazu, ale z zachowaniem ciągłości w obszarach obejmujących trzy punkty widoku rzeczywistego.

W rozdziale 8. opisano dwie zaproponowane techniki korekcji niespójności barwnej: technikę adaptacyjną, dostosowującą się do lokalnych zmian charakterystyki barwnej oraz technikę globalną, działającą w czasie rzeczywistym, kompensującą charakterystykę barwną wielu widoków rzeczywistych w jednakowy sposób dla całego widoku wirtualnego.

W rozdziale 9. opisano technikę poprawy jakości syntezowanych widoków wirtualnych poprzez zwiększenie rozdzielczości widoków rzeczywistych i odpowiadających im map glębi dokonywane przed operacją syntezy widoków wirtualnych.

Rozdział 10. zawiera podsumowanie przeprowadzonych wyników eksperymentalnych. Rozdział jest podzielony na dwie części. W pierwszej części dokonano porównania jakości widoków wirtualnych syntezowanych przy użyciu zaproponowanej metody z widokami syntezowanymi za pomocą oprogramowania odniesienia. W części drugiej zaprezentowano analizę czasu przetwarzania. Opracowaną metodę syntezy porównano z oprogramowaniem odniesienia również w tym aspekcie.

W rozdziale 11. przedstawiono sposób dostosowania opracowanej metody syntezy widoków wirtualnych do pracy w czasie rzeczywistym oraz wpływ poszczególnych zaproponowanych technik na jakość i czas syntezy. Na końcu rozdziału zaprezentowano rezultaty, a więc pomiary jakości widoków wirtualnych syntezowanych przy użyciu zoptymalizowanego algorytmu syntezy.

Rozdział 12. stanowi podsumowanie rozprawy doktorskiej. Zawarto w nim wnioski z przeprowadzonych badań oraz główne osiągnięcia badawcze pracy.

1.6. PUBLIKACJE AUTORA ROZPRAWY

Autor rozprawy jest autorem lub współautorem 46 prac naukowych:

- 2 artykułów w czasopismach o zasięgu międzynarodowym,
- 7 artykułów w czasopismach polskich,
- 37 artykułów w materiałach konferencji międzynarodowych,
 w tym 20 artykułów opublikowanych jako dokumenty grupy ISO/IEC MPEG.

ARTYKUŁY W CZASOPISMACH O ZASIĘGU MIĘDZYNARODOWYM

- 1. O. Stankiewicz, M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, J. Samelak, *A free-viewpoint television system for horizontal virtual navigation*, IEEE Transactions on Multimedia, tom 20, nr 8, s. 2182-2195, 08.2018, indeksowany w Web of Science, Impact Factor: 3.977.
- 2. A. Dziembowski, M. Domański, *View and depth preprocessing for view synthesis enhancement*, International Journal of Electronics and Telecommunications, tom 64, nr 3, s. 269-275, 2018, indeksowany w Web of Science.

ARTYKUŁY W MATERIAŁACH KONFERENCJI MIĘDZYNARODOWYCH INDEKSOWANYCH W BAZACH IEEE XPLORE, SCOPUS I WEB OF SCIENCE

- 3. A. Dziembowski, J. Samelak, M. Domański, *View selection for virtual view synthesis in free navigation systems*, International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018, oczekuje na indeksację.
- 4. A. Dziembowski, J. Stankowski, Real-time CPU-based virtual view synthesis, International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018, oczekuje na indeksację.
- M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, D. Mieloch, O. Stankiewicz, J. Stankowski, K. Wegner, *Real-time virtual navigation provision by simple means*, International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018, oczekuje na indeksację.
- 6. K. Wegner, D. Łosiewicz, T. Grajek, O. Stankiewicz, A. Dziembowski, M. Domański, *Omnidirectional view synthesis and test images*, International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018, oczekuje na indeksacje.
- 7. A. Dziembowski, M. Domański, *Adaptive color correction method in virtual view synthesis*, 3DTV Conference 2018, Sztokholm/Helsinki, Szwecja/Finlandia, 3-5.06.2018, oczekuje na indeksację.
- M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, K. Klimaszewski, D. Mieloch, R. Ratajczak, O. Stankiewicz, J. Siast, J. Stankowski, K. Wegner, *Demonstration of a simple free*

viewpoint television systems, IEEE International Conference on Image Processing, ICIP 2017, Pekin, Chiny, 17-20.09.2017.

- D. Mieloch, A. Dziembowski, A. Grzelka, O. Stankiewicz, M. Domański, *Graph-based multiview depth estimation using segmentation*, IEEE International Conference on Multimedia and Expo, ICME 2017, Hongkong, Chiny, 10-14.07.2017.
- A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, M. Domański, *Enhancing view* synthesis with image and depth map upsampling, International Conference on Systems, Signals and Image Processing, IWSSIP 2017, Poznań, Polska, 22-24.05.2017.
- 11. D. Mieloch, A. Dziembowski, A. Grzelka, O. Stankiewicz, M. Domański, *Temporal* enhancement of graph-based depth estimation method, International Conference on Systems, Signals and Image Processing, IWSSIP 2017, Poznań, Polska, 22-24.05.2017.
- A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, K. Wegner, M. Domański, *Multiview Synthesis – improved view synthesis for virtual navigation*, 32nd Picture Coding Symposium, PCS 2016, Norymberga, Niemcy, 4-7.12.2016.
- A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, M. Domański, *Depth map upsampling and refinement for FTV systems*, International Conference on Signals and Electronic Systems, ICSES 2016, Kraków, Polska, 5-7.09.2016.
- M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, Optimization of camera positions for free-navigation applications, International Conference on Signals and Electronic Systems, ICSES 2016, Kraków, Polska, 5-7.09.2016.
- 15. M. Domański, M. Bartkowiak, A. Dziembowski, T. Grajek, A. Grzelka, A. Łuczak, D. Mieloch, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, *New results in free-viewpoint television systems for horizontal virtual navigation*, IEEE International Conference on Multimedia and Expo, ICME 2016, Seattle, Stany Zjednoczone, 11-15.07.2016.
- A. Dziembowski, M. Domański, A. Grzelka, D. Mieloch, J. Stankowski, K. Wegner, *The influence of a lossy compression on the quality of estimated depth maps*, 23rd International Conference on Systems, Signals and Image Processing, IWSSIP 2016, Bratysława, Słowacja, 23-25.05.2016.
- M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, Ł. Kowalski, M. Kurc, A. Łuczak, D. Mieloch, R. Ratajczak, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, *Methods of high efficiency compression for transmission of spatial representation of motion scenes*, IEEE International Conference on Multimedia and Expo, ICME 2015, Turyn, Włochy, 29.06-3.07.2015.
- M. Domański, A. Dziembowski, D. Mieloch, A. Łuczak, O. Stankiewicz, K. Wegner, *A practical approach to acquisition and processing of free viewpoint video*, 31st Picture Coding Symposium, PCS 2015, Cairns, Australia, 31.05-3.06.2015.

 M. Domański, A. Dziembowski, A. Kuehn, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, *Experiments on acquisition and processing of video for free-viewpoint television*, 3DTV Conference 2014, Budapeszt, Węgry, 2-4.07.2014.

ARTYKUŁY W MATERIAŁACH KONFERENCJI MIĘDZYNARODOWYCH ORGANIZOWANYCH PRZEZ GRUPĘ MPEG (ISO/IEC JTC1/SC29/WG11)

- 20. A. Dziembowski, O. Stankiewicz, [MPEG-I Visual] Fast color correction technique for view synthesis, ISO/IEC JTC1/SC29/WG11 MPEG2018, Ljubljana, Słowenia, M43694, 13-20.07.2018.
- A. Dziembowski, D. Mieloch, K. Wegner, O. Stankiewicz, M. Domański, Proposal of enhanced version of View Synthesis Reference Software with multiple input views, ISO/IEC JTC1/SC29/WG11 MPEG2018, M42941, Ljubljana, Słowenia, 13-20.07.2018.
- M. Domański, D. Łosiewicz, T. Grajek, O. Stankiewicz, K. Wegner, A. Dziembowski, D. Mieloch, *Extended VSRS for 360 degree video*, ISO/IEC JTC1/SC29/WG11 MPEG2018, M41990, Gwangju, Korea Południowa, 22-26.01.2018.
- 23. M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, D. Mieloch, R. Ratajczak, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, *Free-viewpoint television demonstration for sports events*, ISO/IEC JTC1/SC29/WG11 MPEG2018, M41994, Gwangju, Korea Południowa, 22-26.01.2018.
- 24. K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, *Exploration experiments on omnidirectional 6-DoF/3-DoF+ rendering*, ISO/IEC JTC1/SC29/WG11 MPEG2017, M41807, Makau, Chiny, 23-27.10.2017.
- 25. K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, Evaluation of stepin/step-out capability of state-of-the-art view synthesis technology, ISO/IEC JTC1/SC29/WG11 MPEG2017, M40809, Turyn, Włochy, 17-21.07.2017.
- K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, Omnidirectional 6-DoF/3-DoF+ rendering, ISO/IEC JTC1/SC29/WG11 MPEG2017, M40806, Turyn, Włochy, 17-21.07.2017.
- M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, *Experimental video coding software for free navigation applications*, ISO/IEC JTC1/SC29/WG11 MPEG2016, M39527, Chengdu, Chiny, 17-21.10.2016.
- M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, *Coding results for Poznan Fencing 2 and Poznan Blocks* 2 test sequences in free navigation scenario, ISO/IEC JTC1/SC29/WG11 MPEG2016, M39215, Chengdu, Chiny, 17-21.10.2016.

- 29. M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, [FTV AHG] Extended results of Poznan University of Technology proposal for Call for Evidence on free-viewpoint television, ISO/IEC JTC1/SC29/WG11 MPEG2016, M38246, Genewa, Szwajcaria, 30.05-3.06.2016.
- M. Domański, A. Dziembowski, A. Grzelka, A. Łuczak, D. Mieloch, O. Stankiewicz, K. Wegner, *Multiview test video sequences for free navigation exploration obtained using pairs of cameras*, ISO/IEC JTC1/SC29/WG11 MPEG2016, M38247, Genewa, Szwajcaria, 30.05-3.06.2016.
- M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, Study on nonuniform distributions of cameras located on an arc, ISO/IEC JTC1/SC29/WG11 MPEG2016, M38248, Genewa, Szwajcaria, 30.05-3.06.2016.
- 32. M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, [FTV AHG] Technical Description of Poznan University of Technology proposal for Call for Evidence on free-viewpoint television, ISO/IEC JTC1/SC29/WG11 MPEG2016, M37893, San Diego, Stany Zjednoczone, 22-26.02.2016.
- 33. M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, [FTV AHG] Video and depth multiview test sequences acquired with circular camera arrangement – "Poznan Service" and "Poznan People", ISO/IEC JTC1/SC29/WG11 MPEG2015, M36569, Warszawa, Polska, 20-27.06.2015.
- M. Domański, K. Klimaszewski, A. Dziembowski, D. Mieloch, A. Łuczak, O. Stankiewicz, K. Wegner, Freeview navigation (FN) anchor generation using 3D-HEVC with depth for "Poznan Blocks" sequence, ISO/IEC JTC1/SC29/WG11 MPEG2015, M36565, Warszawa, Polska, 20-27.06.2015.
- M. Domański, A. Dziembowski, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, Poznan University of Technology test multiview video sequences acquired with circular camera arrangement – "Poznan Team" and "Poznan Blocks" sequences, ISO/IEC JTC1/SC29/WG11 MPEG2015, M35846, Genewa, Szwajcaria, 14-20.02.2015.
- M. Domański, A. Dziembowski, K. Klimaszewski, A. Łuczak, D. Mieloch, O. Stankiewicz, K. Wegner, *Comments on further standarization for free-viewpoint television*, ISO/IEC JTC1/SC29/WG11 MPEG2015, M35842, Genewa, Szwajcaria, 14-20.02.2015.
- M. Domański, D. Mieloch, A. Dziembowski, O. Stankiewicz, K. Wegner, Super multiview image compression: results for Bee sequence (FTV EE3), ISO/IEC JTC1/SC29/WG11 MPEG2014, M35070, Strasburg, Francja, 20-24.10.2014.
- M. Domański, A. Dziembowski, A. Kuehn, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, *Poznan Blocks – a multiview video test sequence and camera parameters* for free viewpoint television, ISO/IEC JTC1/SC29/WG11 MPEG2014, M32243, San Jose, Stany Zjednoczone, 13-17.01.2014.

 K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, First version of depth maps for Poznan Blocks multiview video test sequence, ISO/IEC JTC1/SC29/WG11 MPEG2014, M32248, San Jose, Stany Zjednoczone, 13-17.01.2014.

ARTYKUŁY W CZASOPISMACH POLSKICH

- 40. A. Dziembowski, J. Stankowski, *Szybka synteza widoków wirtualnych w systemach telewizji swobodnego punktu widzenia*, Przegląd Telekomunikacyjny, nr 6, s. 330-333, 2018.
- 41. A. Dziembowski, A. Grzelka, D. Mieloch, Zwiększanie rozdzielczości obrazu i mapy głębi w celu poprawy jakości syntezy widoków wirtualnych, Przegląd Telekomunikacyjny, tom 86, nr 6, s. 405-408, 2017.
- 42. D. Mieloch, A. Dziembowski, A. Grzelka, *Estymacja glębi dla systemów wielowidokowych*, Przegląd Telekomunikacyjny, tom 86, nr 6, s. 479-482, 2017.
- 43. A. Dziembowski, A. Grzelka, D. Mieloch, Wielowidokowa synteza w systemach telewizji swobodnego punktu widzenia, Przegląd Telekomunikacyjny, tom 88, nr 6, s. 233-236, 2016.
- 44. D. Mieloch, A. Dziembowski, A. Grzelka, *Segmentacja obrazu w estymacji map głębi*, Przegląd Telekomunikacyjny, tom 88, nr 6, s. 241-244, 2016.
- 45. M. Domański, A. Dziembowski, A. Kuehn, D. Mieloch, *Telewizja swobodnego punktu widzenia* – nowa usługa czy futurystyczna wizja?, Przegląd Telekomunikacyjny, nr 8-9, s. 734-737, 2014.
- 46. A. Dziembowski, A. Kuehn, A. Łuczak, D. Mieloch, K. Wegner, Realizacja eksperymentalnego systemu telewizji swobodnego punktu widzenia z łukowym ustawieniem kamer, Przegląd Telekomunikacyjny, nr 6, s. 161-164, 2014.

2. PRZEGLĄD STANU WIEDZY

W rozdziale 2. przedstawiono opisane w literaturze metody syntezy widoków wirtualnych oraz znane techniki uzupełniania widoków wirtualnych, eliminacji wpływu rozmycia krawędzi w widokach rzeczywistych na jakość widoku wirtualnego oraz technik korekcji barwnej. Rozdział ten ma na celu przybliżenie zagadnień potrzebnych do przedstawienia oryginalnych osiągnięć autora w dalszych rozdziałach rozprawy.

2.1. PRZEGLĄD METOD SYNTEZY WIDOKÓW WIRTUALNYCH

Znanych jest wiele metod syntezy widoków wirtualnych, mających różną charakterystykę i znajdujących różne zastosowanie: począwszy od wykorzystania ich w monitorach autostereoskopowych [Lee'13A], poprzez systemy swobodnej nawigacji [Stankiewicz'18], na systemach wirtualnej rzeczywistości [Wegner'18] skończywszy.

Metody syntezy widoków wirtualnych można sklasyfikować na kilka sposobów. Po pierwsze, istniejące metody można podzielić ze względu na ustawienie kamer, do jakiego zostały one stworzone i przystosowane. Wyróżnić możemy metody dostosowane do systemów o konkretnej charakterystyce: systemów z liniowym ustawieniem kamer [Akin'15, Jun-Te'15], kamer pola światła [Kim'13, Levoy'06] albo kamer wszechkierunkowych (tzw. kamer 360°) [Domański'18A, Wegner'17A, Wegner'17C, Wegner'18]. Istnieją również metody pozwalające syntezować widoki wirtualne dla dowolnego ustawienia kamer [Ceulemans'18, Dziembowski'16D, Dziembowski'18A, Mori'09].

Drugim sposobem podziału metod syntezy widoków wirtualnych jest liczba widoków rzeczywistych, jakie są używane do zsyntezowania widoku wirtualnego. Większość metod pozwala na użycie dwóch widoków rzeczywistych [Domański'09A, Du-Hsiu'13, Jin'16, Mori'09, Rogmans'09, Tanimoto'09A]. Istnieją jednak metody, w których widok wirtualny syntezowany jest na podstawie trzech [Akin'15] lub większej liczby widoków rzeczywistych [Ceulemans'18, Cooke'06, Dziembowski'16A, Fachada'18A, Jun-Te'15, Li'18].

Podziału metod syntezy widoków wirtualnych dokonać można również ze względu na czas obliczeń. Wśród nich można wyróżnić metody pozwalające na syntezę widoków wirtualnych w czasie rzeczywistym: zaprojektowane dla układów programowalnych FPGA [Akin'15, Wang'12], kart graficznych [Chen'06, Do'11, Rogmans'09, Yao'16] czy też procesora (CPU, *Central Processing Unit*) [Dziembowski'18A].

Czwartym sposobem klasyfikacji metod syntezy widoków wirtualnych jest podział ze względu na sposób reprezentacji danych wizyjnych. Wśród stosowanych reprezentacji wymienić można przestrzeń promieni (ang. *ray-space*) [Tanimoto'05], stosowaną głównie w systemach z gęstym rozmieszczeniem kamer: kamerach pola światła [Kim'13, Levoy'06] i systemach z liniowym [Fujii'00,

Przegląd stanu wiedzy

Jorissen'14, Yao'13] bądź łukowym [Uemori'08] rozmieszczeniem kamer. Inną reprezentacją jest chmura punktów (ang. *point cloud*), w której poszczególne punkty reprezentujące zarejestrowaną scenę charakteryzowane są poprzez barwę (na przykład w przestrzeni RGB) i położenie we wspólnej przestrzeni trójwymiarowej. Przykładem wykorzystania chmury punktów może być filtracja punktów nadmiarowych [Wei'13] lub posiadających błędną wartość głębi [Ceulemans'18, Skupin'14, Wu'17], tworzenie trójwymiarowych modeli obiektów na podstawie wielu widoków [Holowko'14] czy też lokalizacja i rozpoznawanie obiektów [Bielicki'13, Dubé'18].

Zarejestrowana scena może również być reprezentowana poprzez obiekty [Miller'06, Smolic'05] bądź też siatkę rozpiętą na punktach w trójwymiarowej przestrzeni [Cignoni'08].

Najpopularniejszą reprezentacją jest reprezentacja MVD ("obraz wielowidokowy + głębia", ang. *Multiview Video plus Depth*) [Merkle'07], w której dane wizyjne reprezentowane są poprzez obraz wielowidokowy i głębię, a więc zbiór widoków wraz z odpowiadającymi im mapami głębi. Synteza widoków wirtualnych bazująca na tej reprezentacji jest określana jako DIBR – *Depth Image Based Rendering* [Zinger'10]. MVD jest najpowszechniej stosowaną reprezentacją dla przetwarzania obrazu wielowidokowego – nie tylko w przypadku syntezy widoków wirtualnych, ale i choćby kompresji takiego obrazu [Lafruit'15A] – zarówno dla AVC (*Advanced Video Coding*) [AVC] jak i HEVC (*High Efficiency Video Coding*) [HEVC] powstało wiele rozszerzeń/ulepszeń bazujących właśnie na tej reprezentacji [Domański'13, Domański'14B, Domański'16C, Domański'16E, Domański'16H, Mueller'13, Shao'12, Sullivan'13, Vetro'11].

Ponadto, MVD ma dwie dodatkowe zalety ułatwiające przeprowadzanie badań eksperymentalnych: dostępność materialu testowego (duża liczba zróżnicowanych sekwencji testowych, w tym zarejestrowanych przy użyciu systemów z rzadkim rozmieszczeniem kamer) oraz oprogramowania odniesienia VSRS (*View Synthesis Reference Software*) [Senoh'17A].

Wspomniane zalety reprezentacji MVD spowodowały, iż badania opisane w ramach rozprawy przeprowadzono z wykorzystaniem tej właśnie reprezentacji.

2.2. METODY SYNTEZY WYKORZYSTUJĄCE REPREZENTACJĘ "OBRAZ WIELOWIDOKOWY + GŁĘBIA"

W reprezentacji MVD dane wizyjne są reprezentowane przez widoki rzeczywiste i odpowiadające im mapy glębi (w których każdemu punktowi w widoku rzeczywistym przyporządkowana jest jego glębia, a więc odległość od płaszczyzny przetwornika kamery). Głębia sceny może być zarówno zarejestrowana przez kamery głębi (np. Microsoft Kinect) [Kang'10, Newcombe'11, Um'11], wyznaczona na podstawie stworzonego modelu sceny i znajdujących się w niej obiektów [Debevec'96, Kovacs'15A], jak i estymowana na podstawie punktów widoków rzeczywistych [Gallup'07, Mieloch'16, Mieloch'17C, Senoh'16].

Zgodnie z podziałem przedstawionym w poprzednim punkcie, wśród metod syntezy wykorzystujących reprezentację MVD możemy wyróżnić metody dostosowane do systemów z kamerami ustawionymi liniowo i systemów z dowolnym rozmieszczeniem kamer. Należy zauważyć, iż liniowo ustawione kamery umożliwiają prezentację zarejestrowanej treści na monitorach autostereoskopowych. Zastosowaniem systemów z dowolnym rozmieszczeniem kamer jest umożliwienie widzowi wirtualnej nawigacji w zarejestrowanej scenie.

W systemach liniowych kamery ustawione są tak, by ich osie optyczne były w przybliżeniu równolegle, a wszelkie niedoskonalości ich ustawienia korygowane są poprzez rektyfikację [Cyganek'09, Hartley'15] zarejestrowanych widoków. Dzięki temu, odpowiadające sobie punkty we wszystkich widokach znajdują się w tym samym wierszu różnych obrazów, a sama synteza upraszcza się do poziomego przesunięcia punktów z widoków rzeczywistych (dowolny punkt znajdujący się w wierszu *r* widoku rzeczywistego po przerzutowaniu znajduje się w wierszu *r* widoku wirtualnego). Przykładem takiej metody jest chociażby VSRS-1D [Tanimoto'08A] będący uproszczoną wersją oprogramowania odniesienia do syntezy widoków wirtualnych, czy też metody opisane w [Akin'15, Jun-Te'15]. Metody dostosowane do systemów z liniowym ustawieniem kamer są relatywnie szybkie, ale mało uniwersalne, przez co nie nadają się do praktycznego zastosowania w rzeczywistych implementacjach systemów swobodnej nawigacji.

Metody syntezy widoków wirtualnych dostosowane do dowolnego ustawienia kamer charakteryzują się większą złożonością. Rzutowanie punktów z widoków rzeczywistych jest zazwyczaj realizowane zgodnie ze wzorami (1.3) i (1.4), a więc w celu wyliczenia pozycji i głębi punktu w widoku wirtualnym wektor zawierający pozycję i glębię danego punktu w widoku rzeczywistym mnożony jest przez odwrotną macierz projekcji kamery rzeczywistej i macierz projekcji kamery wirtualnej zgodnie z (1.3), a wynik tej operacji jest następnie dzielony zgodnie z (1.4).

Poszczególne metody różnią się sposobem samego rzutowania. Dla przykładu, syntezę widoku wirtualnego można przeprowadzić w przód (ang. *forward synthesis*) [Do'11, Zinger'10] lub wstecz (ang. *backward synthesis*) [Domański'09A, Du-Hsiu'13, Senoh'17A]. Porównanie obu metod pokazano na rysunku 2.1.

W przypadku syntezy w przód proces rzutowania przebiega jednoetapowo, a barwa punktów widoku wirtualnego rzutowana jest przy użyciu rzeczywistych map głębi (rysunek 2.1B). Synteza wstecz wymaga natomiast przeprowadzenia dwóch operacji rzutowania. W pierwszym kroku tworzona jest wirtualna mapa głębi (mapa głębi, której punkty odpowiadają punktom widoku wirtualnego). W kroku drugim, na podstawie tejże mapy rzutowana jest barwa z widoków rzeczywistych (rysunek 2.1A). W syntezie wstecz jest możliwe przeprowadzenie dodatkowej filtracji wirtualnej mapy głębi mającej na celu usunięcie niewielkich błędów czy też pęknięć (ang. *cracks, crack artifacts*) wynikających z ograniczonej rozdzielczości obrazu i głębi. Filtracja ta jest przeprowadzana przed rzutowaniem barwy

Przegląd stanu wiedzy

z widoków rzeczywistych do widoku wirtualnego. Wadą syntezy wstecz jest jej większa złożoność, albowiem rzutowanie punktów (1.3) musi zostać zrealizowane dwukrotnie: pierwszy raz w celu wyznaczenia pozycji (rys. 1.3) i głębi odpowiadających punktów w wirtualnej mapie głębi i drugi raz w celu ustalenia pozycji w widoku rzeczywistym, z jakiej kopiowany będzie kolor do odpowiadającego mu punktu w widoku wirtualnym.



Rysunek 2.1. Synteza wstecz (A) i w przód (B)

Synteza w przód jest szybsza, jednakże brak możliwości przeprowadzenia filtracji wirtualnej mapy glębi może skutkować większą ilością blędów (np. pęknięć) w widoku wirtualnym, niż w przypadku syntezy wstecz. Z tego powodu większość opisanych algorytmów syntezy widoków wirtualnych wykorzystuje syntezę wstecz. Niemniej jednak, istnieją metody pozwalające na redukcję blędów wynikających z zastosowania syntezy w przód. W pracy [Zinger'10] zaprezentowano dwa rozwiązania pozwalające na redukcję pęknięć w syntezowanym widoku przy użyciu syntezy w przód. Autorzy zaproponowali, aby każdy punkt widoku rzeczywistego był rzutowany do widoku wirtualnego nie jako pojedynczy punkt, ale większy (składający się z kilku sąsiednich punktów) obszar, nazwany przez autorów "wiązką" (ang. *beam*). Rozwiązanie to faktycznie pozwala na usunięcie pęknięć w widoku wirtualnym, jednakże ma jedną podstawową wadę. W przypadku rzeczywistej krawędzi obiektu algorytm zachowa się tak, jak przy uzupełnianiu pęknięć. Tym samym krawędź obiektu zostanie niepoprawnie przesunięta, a on sam poszerzony. Drugim opisanym pomysłem było nadpróbkowanie widoków rzeczywistych, co – kosztem czterokrotnie większej liczby rzutowań – pozwala na redukcję pęknięć.

Interesującą metodę syntezy widoków wirtualnych zaprezentowano w pracy [Lipski'14]. Odpowiadające sobie punkty z dwóch widoków powinny rzutować się w ten sam punkt przestrzeni. Zdanie to jednak nie jest prawdziwe w przypadku niedokładnych parametrów kamer, błędów głębi czy też przesunięcia czasowego przy rejestracji obu widoków. W opisanym podejściu punkty z dwóch widoków rzeczywistych rzutowane są do przestrzeni trójwymiarowej, a następnie estymowana jest odległość pomiędzy nimi w tej przestrzeni. Następnie, algorytm iteracyjnie minimalizuje błąd dla poszczególnych punktów. Opisany algorytm jest niestety bardzo czasochłonny, do przetworzenia jednej ramki sekwencji wymaga nawet do kilku godzin (niestety autorzy pracy [Lipski'14] nie podali konfiguracji użytego komputera), przez co nie nadaje się do zastosowania w rzeczywistym systemie swobodnej nawigacji.

2.3. UZUPEŁNIANIE OBRAZU SYNTETYCZNEGO

Syntezowany widok wirtualny zawiera obszary przerzutowane z widoków rzeczywistych, a także odsłonięcia, czyli obszary, które były zasłonięte (przez bliższe obiekty) zarówno w lewym, jak i prawym widoku rzeczywistym. Podstawowym sposobem wypełniania odsłonięć jest **uzupełnianie** (ang. *inpainting, hole-filling*) tychże obszarów na podstawie zsyntezowanych (nieodsłoniętych) fragmentów widoku wirtualnego.

Z początku, techniki uzupełniania obrazu nie były stosowane do uzupełniania odsłonięć w widokach wirtualnych, a do poprawiania statycznych obrazów (np. zdjęć). W pracy [Bertalmio'00] opisano technikę usuwania niepożądanych elementów obrazu, takich jak nałożony na obraz widoczny znak wodny czy też zadrapania bądź zagięcia na zeskanowanym zdjęciu. W przedstawionej technice obszary te są wypełniane na podstawie najbliższego otoczenia przy zachowaniu kierunku krawędzi obiektów dochodzących do takiego obszaru. Z uwagi na stopniowy sposób zmniejszania uzupełnianych obszarów technika została określona mianem uzupełniania dyfuzyjnego (diffusion-based inpainting). Podobne podejście zastosowano w pracy [Levin'03], przy czym zostało ono rozwinięte o analizę globalnych cech obrazu. Na przykład, jeżeli w obrazie jest wiele ostrych narożników (np. budynków), ewentualny narożnik w obszarze uzupełnionym również będzie ostry; jeżeli większość narożników jest zaokrąglona, uzupełniany narożnik też będzie zaokrąglony. Rozwinięciem techniki opisanej w pracy [Bertalmio'00] jest również algorytm przedstawiony w pracy [Bertalmio'03], w którym uzupełnianie przeprowadzane jest w dwóch krokach: uzupełniania struktury i uzupełniania tekstury. Obraz wejściowy poddawany jest dekompozycji falkowej w celu wydzielenia dwóch składowych: struktury (krawędzie obiektów) i tekstury (charakterystyka poszczególnych obiektów). Do uzupełniania struktury wykorzystywana jest technika opisana w pracy [Bertalmio'00], tekstura uzupełniana jest poprzez wklejanie najlepiej pasujących fragmentów z pozostałej części obrazu, czyli takich obszarów, których otoczenie jest podobne do obszarów otaczających obszar uzupełniany.

Wszystkie trzy wymienione techniki dobrze sprawdzają się w przypadku, gdy uzupełniane obszary mają niewielką szerokość (do kilkudziesięciu okresów próbkowania). Poza tym, techniki te są iteracyjne, powolne, trudne do sprzętowego zaimplementowania, a przez to trudne do wykorzystania w praktycznym systemie swobodnej nawigacji.

Przegląd stanu wiedzy

Technikę, umożliwiającą skuteczne wypełnianie większych obszarów zaprezentowano w pracy [Criminisi'04]. W odróżnieniu od wcześniej wspomnianych technik dyfuzyjnych, bazujących na przeciąganiu koloru z bezpośredniego otoczenia, autorzy zaproponowali uzupełnianie przez wklejanie najbardziej pasujących fragmentów obrazu (*patch-based inpainting*). Dla każdego punktu uzupełnianego obszaru wyszukiwany jest taki fragment pozostalej części obrazu, który jest najbardziej podobny do znanego otoczenia uzupełnianego punktu. Ponadto, stwierdzono, że kolejność uzupełniania punktów w uzupełnianym obszarze ma bardzo duże znaczenie. W pracy [Criminisi'04] zdecydowano, aby podczas uzupełniania obszaru brać pod uwagę krawędzie dochodzące do uzupełnianego obszaru i punkty leżące na przedłużeniu tychże krawędzi uzupełniać w pierwszej kolejności, w praktyce dzieląc uzupełniany obszar na części. Podobnie jak technika przedstawiona w pracy [Bertalmio'00], również algorytm opisany w pracy [Criminisi'04] powstał wcześniej, niż zaczęto myśleć o systemach telewizji swobodnego punktu widzenia, jednakże znajduje on zastosowanie również w tej dziedzinie [Tezuka'14].

Rozwinięcie techniki zaprezentowanej w pracy [Criminisi'04] opisano w pracy [Kumar'05], gdzie pod uwagę wzięto również spójność czasową wypełnianych obszarów, a najbardziej pasujące fragmenty są wyszukiwane nie tylko w tym samym obrazie, ale również w sąsiednich ramkach.

Zarówno w pracy [Criminisi'04], jak i pracy [Kumar'05] wyszukiwanie najbardziej pasujących fragmentów obrazu jest realizowane poprzez porównywanie wartości próbek w prostokątnym oknie. W pracy [Schmeing'15] prostokątne okno zostało zastąpione oknem o nieregularnym kształcie, wydzielonym poprzez wcześniejszą segmentację obrazu wejściowego. Podejście takie wymaga przeprowadzenia większej liczby operacji, jednakże pozwala lepiej dopasować się do charakterystyki obrazu.

Inną techniką uzupełniania bazującą na wyszukiwaniu najbardziej pasujących obszarów w pozostałej części obrazu jest opisana w pracy [Barnes'09] technika *PatchMatch*. W technice tej postawiono na szybkość działania, rezygnując z metodycznego przeszukiwania obrazu w celu znalezienia podobnych obszarów. Zamiast tego autorzy zaproponowali, by początkowa pozycja potencjalnie pasującego obszaru była losowana, a następnie iteracyjnie poprawiana.

Technika opisana w pracy [Barnes'09] została wykorzystana w pracy [Huang'14], gdzie dodano istotne usprawnienie, a mianowicie wykrywanie płaszczyzn w obrazie. Wydzielone z obrazu płaszczyzny są następnie rektyfikowane, aby linie równoległe w rzeczywistości były równoległe także na obrazie. Poszukiwanie najbardziej pasujących fragmentów odbywa się w zrektyfikowanych obrazach, a tym samym zmniejszany jest wpływ perspektywy na uzupelnianie.

Do uzupełnienia obrazu można zastosować także metody optymalizacyjne [Komodakis'07, Habigt'13, Ceulemans'16], na przykład metodę propagacji wierzeń (ang. *Belief Propagation*) [Pearl'82].

Podejście takie jest często rozwiązaniem czasochłonnym – w przypadku algorytmu opisanego w pracy [Komodakis'07] jedna ramka sekwencji *BBB Flowers* przetwarzana jest około 20 minut, dla algorytmu z pracy [Ceulemans'16] jest to (pomimo zastosowanych uproszczeń) kilkadziesiąt sekund na ramkę. W pracy [He'12] opisano statystyczną analizę kierunku kopiowania najbardziej podobnych obszarów. Jak wykazano, znając najczęstsze kierunki kopiowania można użyć metod optymalizacyjnych (chociażby metody cięcia grafu, ang. *Graph Cuts*) w celu minimalizacji funkcji celu, a wskutek tego zwiększyć jakość uzupełnianych obszarów.

Innym interesującym rozwiązaniem jest uzupełnianie hierarchiczne. W takim podejściu rozdzielczość widoku wirtualnego jest zmniejszana do momentu, gdy nie ma w nim żadnych odsłonięć. Następnie, kolor punktów z obrazu o mniejszej rozdzielczości jest używany do wypełniania odsłonięć w widoku o rozdzielczości dwukrotnie większej aż do momentu uzupełnienia widoku wejściowego. Najprostsze podejście bazujące na tej idei przedstawiono w pracach [Solh'12] i [Lu'13]. W pracy [LeMeur'13] widoki wirtualne o różnych rozdzielczościach są łączone przy użyciu metod optymalizacyjnych.

2.3.1. Uzupełnianie z uwzględnieniem głębi

Wszystkie wyżej wspomniane techniki uzupełniania obrazu nie uwzględniają dodatkowej informacji, jaka jest dostępna w przypadku syntezy widoków, a mianowicie głębi. W celu poprawnego uzupełnienia odsłonięć głębia powinna zostać uwzględniona, a odsłonięcia powinny zostać uzupełnione na podstawie obszarów odpowiadających dalszym obiektom (np. tłu, które było przysłonięte w widokach rzeczywistych przez znajdujące się bliżej obiekty). Tym samym do uzupełniania odsłonięć należy użyć wyłącznie informacji z obszarów posiadających większą wartość głębi.

Prostą technikę takiego uzupełniania obrazu opisano w pracy [Oh'09], gdzie sprawdzano, czy dany obszar odsłonięty graniczy zarówno z obszarami dalszymi (tłem), jak i bliższymi. W takim przypadku te z punktów bezpośrednio otaczających odsłonięty obszar, których głębia jest mniejsza są zastępowane punktami reprezentującymi dalsze obiekty. Następnie, gdy cały obszar odsłonięty otoczony jest punktami tła wykonywane jest uzupełnianie dyfuzyjne.

Równie prostą technikę opisano w pracy [Zinger'10], gdzie dla każdego punktu z odsłoniętego obszaru analizowana jest głębia najbliższych zsyntezowanych punktów w 8 głównych kierunkach (najbliższego przerzutowanego punktu z lewej i prawej strony, z góry i z dołu, a także czterech punktów położonych po ukosie pod kątem 45 i 135 stopni). Analizowany punkt wypełniany jest przy użyciu średniej ważonej barw tych z 8 najbliższych punktów, które nie reprezentują bliskich obiektów. Podobną technikę wykorzystano w oprogramowaniu odniesienia VSRS [Senoh'17A], w którym również analizowana jest głębia ośmiu najbliższych zsyntezowanych punktów. W odróżnieniu od techniki opisanej w pracy [Zinger'10], gdzie punkty w odsłoniętym obszarze są uzupełniane niezależnie

Przegląd stanu wiedzy

od siebie, w VSRS uzupełniane są całe wiersze tychże obszarów. Dla przykładu, jeżeli obszar odsłonięty miał szerokość 10 okresów próbkowania, w oprogramowaniu VSRS [Senoh'17A] analizowane będzie otoczenie dziesięciu punktów i wszystkie te punkty zostaną uzupełnione tą samą wartością.

W pracach [Daribo'10], [Buyssens'15] i [Buyssens'17] przedstawiono rozwinięcie techniki opisanej w pracy [Criminisi'04]. Zgodnie z obserwacją, że lepiej jest uzupelnić obszar odsłonięty informacją z tła, a nie punktów bliskich, do uzupelniania odsłonięć wykorzystywane są głównie te punkty, które charakteryzują się większą głębią.

W pracy [Cho'17] rozwinięto technikę opisaną w pracy [Barnes'09]. Rozwinięcie techniki polega na dodatkowym uwzględnieniu glębi podczas przeszukiwania obrazu celem wyszukania obszarów, których treść będzie mogła zostać skopiowana do uzupełnianego obszaru.

Informację o głębi wykorzystano również w pracach [Ahn'13] i [Ndjiki-Nya'11]. Opisane tam algorytmy uzupełniania działają dwuetapowo. W pierwszym kroku uzupełniana jest wirtualna mapa głębi (również w tych technikach punkty dalsze mają priorytet przy uzupełnianiu). Dodatkowo, w pracy [Ndjiki-Nya'11] wirtualna mapa głębi jest filtrowana w celu usunięcia błędów spowodowanych rozmyciem krawędzi w wejściowych mapach głębi. Następnie do uzupełniania punktów w widoku wirtualnym wykorzystywane są te punkty z pozostałej części obrazu, które mają najbardziej pasujące otoczenie. W odróżnieniu od pracy [Criminisi'04] i innych technik uzupełniania przez wklejanie, podczas wyszukiwania najbardziej podobnych obszarów w obu technikach uwzględniana jest informacja o głębi.

W niektórych pracach skupiono się na szczególnych przypadkach występowania odsłonięć w widoku wirtualnym. Dla przykładu, w pracach [Mao'13] i [Mao'14] opisano optymalizacyjny algorytm mający na celu usunięcie w widoku wirtualnym nieciągłości powstających podczas wirtualnego zbliżania się do zarejestrowanej sceny. W widoku wirtualnym powstaje w takim przypadku wiele obszarów bez przerzutowanej informacji, jako że widok wirtualny ma tę samą rozdzielczość, co widoki rzeczywiste, a przedstawia mniejszy niż one wycinek rejestrowanej sceny. W pracy [Koeppel'15] zaprezentowano technikę hybrydowego uzupełniania odsłonięć w ekstrapolowanych widokach wirtualnych, w której uzupełnianie podzielone jest (podobnie jak w pracy [Bertalmio'03]) na proces uzupełniania struktury (krawędzi obiektów) i tekstury.

Inne rozwiązanie przedstawiono w pracy [Wang'11], a zoptymalizowano w pracach [Plath'13] i [Khatiullin'18]. W przedstawionej technice odsłonięcia usuwane są poprzez przestrzenne rozciąganie tych obszarów tła, które znajdują się w pobliżu niezsyntezowanego obszaru.

Inne podejście do zmniejszania powierzchni odsłonięć przedstawiono w pracach [Liu'12], [Luo'17] i [Yai'14]. W technikach tych autorzy zaproponowali wyznaczenie tła (zarówno jego głębi, jak
i barwy) w celu wyznaczenia, co dokładnie zostało odsłonięte przez bliższe obiekty. W takim podejściu, jeżeli obiekt znajdujący się bliżej odsłoni fragment tła, możliwe będzie uzupełnienie tego obszaru wyznaczonym tłem.

W pracy [Luo'17] opisano wyznaczenie tła poprzez usuniecie tych obszarów widoku wirtualnego, które charakteryzują się niewielką głębią. Aby uniknąć usunięcia bliższych fragmentów tła, jak choćby części ustawionej pod kątem ściany (np. w sekwencji Ballet [Zitnick'04]), usuwane obszary muszą być oddzielone od otoczenia ostrą krawędzią w mapie głębi. Z powodu niewystarczającej jakości map głębi, w przedstawionym algorytmie głębia musi zostać przefiltrowana. Aby uniknąć rozmycia krawędzi, jakie wprowadziłaby filtracja dolnoprzepustowa, zastosowano filtr bilateralny zachowujący ostre krawędzie w miejscach, gdzie krawędź głębi pokrywa się z krawędzią w widoku a wygładzający głębię w miejscach, gdzie w widoku krawędzi nie ma. Po usunięciu bliższych obiektów te obszary, w których się one znajdowały muszą zostać uzupełnione. W większości przypadków głębia punktów w tych obszarach jest interpolowana w sposób liniowy pomiędzy najbliższym lewym i prawym punktem tła. Jeżeli różnica głębi najbliższego lewego i prawego punktu jest zbyt duża, punkt wypełniany jest głębią skopiowaną z dalszego z nich. Takie podejście generuje blędy, chociażby w przypadku, gdy w uzupełnianym obszarze nie ma wyłącznie jednej, a dwie zbiegające się powierzchnie (np. kąt pomieszczenia w sekwencji Ballet). W celu eliminacji błędów wypełnione obszary są później iteracyjnie optymalizowane, jednakże przekłada się to na zwiększony czas obliczeń, a więc brak możliwości użycia techniki w rzeczywistym systemie.

W pracy [Liu'12] zaprezentowano metodę GBVS (*Global-background based view synthesis*). Przy użyciu informacji z wielu ramek sekwencji tworzone jest globalne tło widoku wirtualnego. W celu stworzenia tła, do widoku wirtualnego są rzutowane wyłącznie nieruchome obiekty. Następnie poszczególne ramki zawierające tło widoku wirtualnego są łączone poprzez uśrednianie. Dla zmniejszenia czasu obliczeń, wykorzystywana jest wyłącznie część ramek, tzw. ramki kluczowe. Algorytm jest bardzo prosty, aczkolwiek skuteczny w przypadku ruchomych obiektów. Jedyny problem stanowią nieruchome bliskie obiekty. Jeżeli dany obiekt się nie poruszył to należy on do tła, niezależnie od głębi, a więc obszar przysłonięty przez taki obiekt musi zostać uzupełniony w inny sposób. Ze względu na szybkość zastosowano w tym celu uzupełnianie dyfuzyjne opisane w pracy [Telea'04].

Podobne podejście opisano w pracach [Yao'14] i [Lai'17], gdzie również na podstawie wielu ramek sekwencji wyznaczane jest statyczne tło widoku wirtualnego. W celu jego wyznaczenia analizowano, jak zmienia się głębia poszczególnych punktów w czasie. Podobnie jak w pracy [Liu'12], kolor punktów tła estymowany jest poprzez uśrednienie wielu ramek, przy czym w przedstawionym algorytmie uśredniany jest kolor wyłącznie z tych ramek, dla których głębia punktu jest duża, a więc

Przegląd stanu wiedzy

punkt reprezentuje tło. Pozostałe, nieuzupełnione informacją z tła, obszary uzupełniane są poprzez kopiowanie podobnych obszarów opisane w pracy [Criminisi'04].

Również w pracy [Zhu'13] przedstawiono wykorzystanie informacji o tle, przy czym informacja ta nie jest używana do redukcji powierzchni odsłoniętych obszarów, a do samego uzupełniania. W opisanej metodzie podczas rzutowania punktów do widoku wirtualnego zapamiętywane są nie tylko punkty z najmniejszą głębią, ale również te, które w widoku wirtualnym zostały przysłonięte bliższymi obiektami. Informacja z przysłoniętych obszarów jest następnie używana do dyfuzyjnego uzupełniania odsłonięć, umożliwiając uzyskanie lepszej jakości uzupełniania w przypadku obszarów odsłoniętych graniczących z obiektem przysłaniającym (dana jest informacja o kolorze i głębi punktów przysłanianych i to ich można użyć do uzupełniania).

W pracy [Zhu'16A] przedstawiono udoskonalenie pomysłu z pracy [Zhu'13]. Proces rzutowania punktów do widoku wirtualnego podzielono na dwa etapy. W pierwszym kroku do poszczególnych punktów widoku wirtualnego rzutowane są te punkty z widoków rzeczywistych, które po przerzutowaniu mają największą głębię. Następnie tak zsyntezowany widok jest uzupełniany przy użyciu techniki przedstawionej w pracy [Criminisi'04]. Podczas drugiego rzutowania punkty z widoków rzeczywistych rzutowane są raz jeszcze na wcześniej uzupełniony widok, przy czym tym razem do widoku wirtualnego zapisywane są punkty bliższe.

2.3.2. UZUPEŁNIANIE Z UWZGLĘDNIENIEM INFORMACJI Z DALSZYCH WIDOKÓW RZECZYWISTYCH

Przedstawione w poprzednim punkcie metody syntezy mają jedną podstawową wadę. Żadna z nich nie wykorzystuje podstawowej informacji, która dostępna jest w przypadku sekwencji wielowidokowych czyli **informacji z innych widoków** rzeczywistych niż tylko 2 sąsiednie. Obszar widoku wirtualnego niewidoczny w najbliższym lewym i prawym widoku mógł być bowiem widoczny w innych widokach.

Syntezę z większej liczby widoków rzeczywistych zaprezentowano w pracach [Ceulemans'18, Cooke'06, Dziembowski'16A, Dziembowski'18G, Fachada'18A, Li'18, Zhu'16B].

W metodzie opisanej w pracy [Cooke'06] punkty z każdego widoku rzeczywistego rzutowane są do osobnego pośredniego widoku wirtualnego. W każdym z nich liczone jest lokalne zagęszczenie przerzutowanych punktów (dla przykładu, gdy 20 punktów z danego widoku rzeczywistego zostało przerzutowane na mały obszar widoku wirtualnego, zagęszczenie przerzutowanych punktów w tymże obszarze jest duże). Następnie wszystkie pośrednie widoki wirtualne łączone są w taki sposób, by dla wszystkich fragmentów połączonego widoku wybrać odpowiadający fragment z tego widoku pośredniego, w którym lokalne zagęszczenie było największe. W pracy [Cooke'06] wzięto pod uwagę również problem wzajemnego przysłaniania się obiektów i w przypadku wyboru spośród kandydatów

o różnej głębi wybierane były obszary leżące bliżej (nawet jeżeli charakteryzowały się one mniejszym zagęszczeniem punktów). W podejściu tym nie wzięto jednak pod uwagę dwóch podstawowych problemów sekwencji wielowidokowych: niepoprawnych wartości map głębi i różnej charakterystyki barwnej poszczególnych widoków. Punkty widoku wirtualnego są wprost kopiowane z widoków rzeczywistych, co skutkuje licznymi błędami pogarszającymi jakość syntezowanych widoków wirtualnych.

W pracy [Ceulemans'18] opisano metodę syntezy wielowidokowej, w której używane są cztery najbliższe widoki rzeczywiste. Obrazy z dalszych kamer systemu są pomijane, by uniknąć błędów wynikających z próbkowania obrazu czy też różnej charakterystyki oświetlenia w różnych kamerach. W opisanej metodzie punkty z wszystkich czterech widoków rzeczywistych są rzutowane do widoku wirtualnego. Jeżeli do danego punktu widoku wirtualnego przerzutowano punkty z więcej niż jednego widoku rzeczywistego (1.7), wybierany jest ten z kandydatów, który znajduje się najbliżej kamery wirtualnej, niezależnie z którego widoku rzeczywistego pochodził. Jeżeli głębia punktu przerzutowanego ze wszystkich widoków jest podobna, barwa punktu wyliczana jest przez uśrednienie barwy przerzutowanej z dwóch najbliższych widoków. Opisana w pracy [Ceulemans'18] metoda ma szereg niedoskonałości. Po pierwsze, całkowicie pomijane są w niej dalsze widoki rzeczywiste. Drugim problemem jest bezkrytyczny wybór punktu przerzutowanego z dalszego widoku rzeczywistego, jeżeli jego głębia była mniejsza od głębi przerzutowanej z innych kamer. Podejście takie jest bardzo wrażliwe na niepoprawnie wyznaczone mapy głębi. Błędy w głębi skutkują pojawianiem się w syntezowanym widoku nieistniejących obiektów. Innym skutkiem wybierania punktów o najmniejszej głębi może być poszerzenie bliskich obiektów (przesunięcie ich krawędzi) spowodowane ograniczoną rozdzielczością widoków rzeczywistych.

Inną metodę syntezy widoków wirtualnych używającą większej liczby widoków rzeczywistych opisano w pracy [Fachada'18A]. Opisana metoda pozwala na syntezę widoku wirtualnego na podstawie nieograniczonej liczby widoków rzeczywistych. W zaproponowanym przez autorkę podejściu barwa punktów widoku wirtualnego wyznaczana jest poprzez zmieszanie barwy punktów przerzutowanych ze wszystkich widoków rzeczywistych. Podejście takie dobrze sprawdza się w przypadku sekwencji zarejestrowanych przy użyciu systemów z liniowym rozmieszczeniem kamer (np. [Goorts'14B]), czy też macierzy równolegle skierowanych kamer (np. [Bonatto'17, Fachada'18B]), gorzej radząc sobie z sekwencjami zarejestrowanymi za pomocą systemów z rzadkim, nieliniowym rozmieszczeniem kamer (np. [Domański'16F]). Podejście opisane w pracy [Fachada'18A] nie pozwala również na uzyskanie poprawnej jakości syntezowanych widoków w przypadku map glębi zawierających blędy (np. [Domański'14C, Goorts'14C, Zitnick'04]), gdyż barwa punktów przerzutowanych blędnie choćby z jednego widoku wpływa na barwę punktów w widoku wirtualnym. Drugą cechą metody opisanej w pracy [Fachada'18A] jest zmodyfikowany sposób rzutowania punktów z widoków rzeczywistych do widoku wirtualnego. W zaproponowanym przez autorkę podejściu sąsiednie punkty widoku

Przegląd stanu wiedzy

rzeczywistego nie są rzutowane niezależnie co skutkuje zachowaniem ciągłości obiektów w widoku wirtualnym. W przypadku bardziej skomplikowanej treści podejście to może jednak skutkować niepoprawnym łączeniem różnych obiektów w widoku wirtualnym.

Syntezę z wielu kamer zaprezentowano również w pracach [Li'18] i [Zhu'16B]. W pokazanym podejściu widok wirtualny syntezowany jest z dwóch najbliższych widoków rzeczywistych, a pozostałe widoki (od 2 do 4) wykorzystywane są wyłącznie do wypełniania odsłonięć. Podejście takie znacząco redukuje powierzchnię obszarów odsłoniętych (według autorów powierzchnia obszarów odsłoniętych jest od 30 do 80% mniejsza, w zależności od sekwencji testowej). W celu zwiększenia szybkości działania algorytmu, z dalszych kamer nie są rzutowane wszystkie punkty, a jedynie te należące do tła. Ma to tę wadę, iż w przypadku przysłaniania w sąsiednich widokach rzeczywistych obiektu mającego większą głębię niż obiekty najbliższe, ale mniejszą niż tło w widoku wirtualnym nie zostanie on przerzutowany z dalszych kamer, a obszar, w którym się on znajduje zostanie błędnie uzupełniony. Poza tym, opisany algorytm jest wrażliwy na błędy w mapach głębi, nie jest bowiem sprawdzana spójność międzywidokowa punktów rzutowanych z dalszych widoków. Jeżeli więc punkt uzupełnianego obszaru może zostać przerzutowany z pozostałych widoków rzeczywistych, zostanie on przerzutowany, niezależnie od tego czy barwa i głębia rzutowanego punktu zgadza się we wszystkich widokach. Opisane niedoskonałości algorytmu skutkują dla niektórych treści (obecnych między innymi w sekwencjach testowych) gorszą jakością syntezowanych widoków w porównaniu z tradycyjnym uzupełnianiem, np. dla sekwencji Breakdancers [Zitnick'04] wypełnianie odsłonięć przy użyciu dalszych widoków rzeczywistych zmniejsza wartość PSNR uzupełnionych obszarów o 2 dB w porównaniu z typowym uzupełnianiem odsłonięć.

Jak pokazano powyżej, znane metody syntezy wykorzystujące informację z dalszych widoków rzeczywistych posiadają szereg wad zmniejszających jakość syntezowanych widoków wirtualnych, takich jak założenie poprawności map głębi i wyrównanej charakterystyki barwnej poszczególnych widoków.

2.4. ELIMINACJA WPŁYWU ROZMYCIA KRAWĘDZI W WIDOKACH RZECZYWISTYCH NA JAKOŚĆ SYNTEZY

Odsłonięcia nie są jedyną przyczyną błędów w widokach wirtualnych. Innym zjawiskiem powodującym powstawanie błędów w syntezowanych widokach jest **rozmycie krawędzi obiektów** w widokach rzeczywistych. Skutkuje to pojawianiem się w widoku wirtualnym nieistniejących w rzeczywistości krawędzi i obiektów.

Problem usuwania nieistniejących krawędzi w wirtualnym widoku został opisany w licznych pracach. W większości z nich błędy te eliminowane są w podobny sposób: punkty odznaczające się dużą głębią graniczące z punktami o obszarami o głębi mniejszej (a więc punkty leżące na krawędzi tła

i bliższych obiektów) nie są w ogóle rzutowane do widoku wirtualnego [Ahn'13] lub są rzutowane wyłącznie wtedy, gdy dany punkt widoku wirtualnego nie został przerzutowany z żadnego innego widoku rzeczywistego [Ceulemans'18, Lee'08, Mueller'08].

Inną technikę eliminacji nieistniejących krawędzi zaproponowano w pracy [Zhao'11]. Zamiast pomijać przy rzutowaniu wszystkie punkty tla graniczące z bliższymi obiektami, autorzy zdecydowali się analizować krawędzie widoku rzeczywistego i odpowiadającej mu mapy glębi. Jeżeli bliski obiekt jest "szerszy" w widoku wirtualnym niż w mapie glębi, a więc występuje przesunięcie jego krawędzi pomiędzy tymi dwoma obrazami, krawędź w mapie glębi jest przesuwana. Punkty leżące przy krawędzi nie są rzutowane jedynie w przypadku, gdy krawędź w widoku rzeczywistym jest rozmyta.

W bardziej zaawansowanej technice opisanej w pracy [Yang'10] przy rzutowaniu do widoku wirtualnego nie są pomijane wszystkie punkty leżące na krawędzi tła i bliskich obiektów, a jedynie te, które autorzy określają jako niewiarygodne. Wiarygodność poszczególnych punktów w lewym widoku rzeczywistym sprawdzana jest poprzez rzutowanie wszystkich punktów lewego widoku do widoku prawego traktowanego jako widok odniesienia. Analogicznie, dla wyznaczenia wiarygodności punktów widoku prawego wszystkie punkty tego widoku rzutowane są do lewego widoku rzeczywistego. Znaczna różnica jasności punktu rzutowanego i znajdującego się w tym samym miejscu punktu widoku odniesienia skutkuje określeniem danego punktu jako niewiarygodny.

Opisane w pracy [Yang'10] rozwiązanie zaproponowano jednak dla prostych przypadków z dwoma liniowo ustawionymi kamerami. Z powodu znacznych różnic pomiędzy różnymi widokami rzeczywistymi, takie podejście nie sprawdziłoby się w przypadku wielokamerowego systemu z rzadko rozmieszczonymi kamerami.

Poza techniką opisaną w pracy [Yang'10], wszystkie wymienione techniki eliminujące wpływ rozmytych krawędzi w widokach rzeczywistych na jakość widoku wirtualnego mają jedną podstawową wadę: punkty znajdujące się przy krawędziach nie są w nich rzutowane nawet wtedy, gdy ich barwa jest poprawna.

2.5. KOREKCJA BARWNA

Ostatnim zjawiskiem pogarszającym jakość syntezowanych widoków jest zróżnicowane oświetlenie sceny w różnych widokach rzeczywistych. Zjawisko to jest widoczne głównie dla sekwencji zarejestrowanych przy pomocy systemów wielokamerowych z rzadkim rozmieszczeniem kamer [Dziembowski'18C].

Istnieje wiele technik korekcji barwnej, które to techniki można podzielić na parametryczne i nieparametryczne [Xu'10]. Wśród technik parametrycznych, zakładających zależność między przestrzeniami barwnymi porównywanych obrazów, można wyróżnić technikę przeniesienia koloru

Przegląd stanu wiedzy

[Reinhard'01]. Najbardziej rozpowszechnioną techniką nieparametryczną jest technika dopasowania histogramów [Fecker'08]. Techniki te nie mogą jednak być wprost użyte do korekcji widoków w rzadkich systemach wielokamerowych z powodu stosunkowo niewielkiego wspólnego obszaru sceny zarejestrowanego przez różne kamery.

Dla przykładu, w technice bazującej na dopasowaniu histogramu dokonywane jest porównanie i korekcja histogramu wyznaczanego dla całego obrazu. W przypadku, gdy w jednym widoku znajduje się obiekt niezarejestrowany w widoku drugim (w którym dany obiekt mógł być na przykład przysłonięty), obiekt ten zmniejsza korelację pomiędzy oboma histogramami, uniemożliwiając poprawną korekcję.

Znane są również techniki uwzględniające różnice pomiędzy widokami. W tych technikach różnica charakterystyki poszczególnych widoków wyznaczana jest jedynie dla obszaru wspólnego. Wśród technik takich nie sposób nie wspomnieć o technice opisanej w pracy [Yamamoto'07], w której przy użyciu techniki SIFT (*Scale Invariant Feature Transform*) [Lowe'04] wyznaczane są odpowiadające sobie punkty w różnych widokach. Dla punktów tych jest sprawdzana różnica intensywności dla wszystkich trzech składowych RGB, a następnie jest wyznaczana (niezależna dla każdej składowej) funkcja przeksztalcająca charakterystykę barwną poprawianego obrazu w charakterystykę obrazu odniesienia. Sama idea korekcji jest więc podobna, jak w przypadku [Reinhard'01].

SIFT stosowany jest również w technice opisanej w pracy [Fezza'14], gdzie zaproponowano technikę pozwalającą zmniejszyć strumień bitowy w przypadku kodowania sekwencji wielowidokowych. Podobnie jak w przypadku algorytmu opisanego w pracy [Yamamoto'07], charakterystyka barwna porównywanych obrazów jest estymowana dla wyznaczonych przy użyciu SIFT odpowiadających sobie punktów, przy czym w technice [Fezza'14] brane pod uwagę jest również najbliższe otoczenie tychże. Następnie charakterystyka poprawianego obrazu jest korygowana poprzez dopasowanie histogramu do histogramu obrazu odniesienia.

Technika dopasowania histogramu została użyta również w algorytmie opisanym w pracy [Lu'15]. Do wykrycia odpowiadających sobie punktów użyto deskryptora SURF (*Speeded-Up Robust Feature*) [Bay'08], dodatkowo sprawdzając spójność w czasie odpowiadających sobie punktów. Następnie dla wykrytych punktów i ich otoczenia wyznaczane są histogramy, by w ostatnim kroku dopasować charakterystykę poprawianego widoku na podstawie dopasowania histogramu do histogramu widoku odniesienia.

W pracy [Shao'08] zaproponowano, by korekcję barwną przeprowadzać osobno dla tła i obiektów znajdujących się bliżej, a odpowiadających sobie punktów szukać poprzez pasowanie bloków z odjętą średnią wartością (MRSAD, *Mean-Removed Sum of Absolute Differences*). Następnie, przy zastosowaniu regresji liniowej, zarówno dla tła, jak i bliższych obiektów, wyznaczana jest funkcja przekształcająca barwę punktów z poprawianego widoku zgodnie z charakterystyką widoku odniesienia.

W pracy [Ye'17] zwrócono uwagę, iż technika dopasowania histogramu nie nadaje się do korekcji koloru w przypadku rzadziej rozmieszczonych kamer. Zamiast tego zaproponowano metodę optymalizacyjną, w której minimalizowana jest energia trzech składników, odpowiadających za zachowanie gradientu, spójność czasową i przestrzenną, a także zachowanie struktury oryginalnego obrazu. Opisane rozwiązanie jest jednak czasochlonne i wymaga (według autorów) 20 sekund na przetworzenie jednej ramki wysokiej rozdzielczości sekwencji.

Wszystkie wymienione techniki pozwalają dopasować charakterystykę barwną poszczególnych widoków do jednego, wybranego widoku odniesienia. Nie jest to jednak dobre podejście w przypadku systemu swobodnej nawigacji, w którym powinno się skompensować **niespójność barwną** różnych widoków przy jednoczesnym zapewnieniu widzowi wrażenia naturalności oglądanej sceny, w tym również naturalnej zmiany oświetlenia podczas wirtualnego przemieszczania się. Techniki umożliwiające korekcję niespójności barwnej różnych widoków opracowano w ramach rozprawy i zaprezentowano w rozdziale 8.

2.6. PODSUMOWANIE

W rozdziale dokonano przeglądu rozwiązań dotyczących syntezy widoków wirtualnych i technik rozwiązujących problemy typowe dla syntezy widoków w systemach wielokamerowych, takie jak uzupełnianie odsłonięć, korekcja barwna czy eliminacja rozmytych krawędzi obiektów. Jak pokazano, istniejące rozwiązania nie rozwiązują wszystkich problemów wynikających z umożliwienia widzowi swobodnej nawigacji w systemie z rzadkim rozmieszczeniem kamer.

W punktach 2.1 i 2.2 sklasyfikowano i przedstawiono znane metody syntezy widoków wirtualnych. W punkcie 2.3 skupiono się na technikach uzupełniania odsłonięć, wliczając w to techniki uwzględniające informację o głębi (punkt 2.3.1) i uzupełnianie przy użyciu dalszych widoków rzeczywistych (punkt 2.3.2). Zauważono, iż istniejące metody mają wady ujawniające się w przypadku syntezy widoków wirtualnych w systemach z rzadkim rozmieszczeniem kamer.

W punkcie 2.4 opisano znane techniki zmniejszające wpływ rozmycia krawędzi w widokach rzeczywistych na jakość widoku wirtualnego. Zauważono, iż znane techniki mają niedoskonałości redukujące jakość syntezowanego widoku wirtualnego.

W punkcie 2.5 scharakteryzowano znane techniki korekcji barwnej, skupiając się na ich przydatności do korekcji widoku wirtualnego. Zauważono, iż techniki te pozwalają na korekcję barwną widoków rzeczywistych przed syntezą, jednakże nie umożliwiają korekcji niespójności barwnej punktów rzutowanych z różnych widoków rzeczywistych podczas syntezy widoku wirtualnego. Tym

Przegląd stanu wiedzy

samym, przy użyciu znanych technik nie jest możliwe zapewnienie naturalnej zmiany oświetlenia podczas wirtualnego przemieszczania się w scenie.

3. METODOLOGIA BADAŃ

3.1. WPROWADZENIE

Syntezowane widoki wirtualne charakteryzują się szeregiem niedoskonałości, spowodowanych między innymi przysłonięciami, niepoprawnymi mapami głębi, zróżnicowanym oświetleniem sceny w różnych widokach rzeczywistych, czy też ograniczoną rozdzielczością widoków rzeczywistych.

Niestety, nie istnieją modele teoretyczne pozwalające oszacować jakość widoków wirtualnych przy użyciu różnych algorytmów syntezy widoków. Z tego powodu, opracowana w ramach rozprawy metoda syntezy widoków wirtualnych została zaimplementowana i sprawdzona w badaniach eksperymentalnych na materiale testowym.

Jakość widoków wirtualnych syntezowanych przy użyciu zaproponowanej przez autora rozprawy i opisywanej w rozprawie metody MVS (*MultiView Synthesis*) została zbadana z wykorzystaniem dwunastu zróżnicowanych sekwencji testowych (punkt 3.4). Ocenie została poddana jakość mierzona w sposób zarówno obiektywny (punkt 3.6.1), jak i subiektywny (punkt 3.6.2).



PSNR, SSIM obliczone dla widoku wirtualnego i

Rysunek 3.1. Sposób estymacji jakości syntezowanego widoku względem rzeczywistego widoku i

Metodologia badań

Do zmierzenia jakości obiektywnej użyto dwóch miar: PSNR i SSIM [Wang'04]. Jakość estymowana była zgodnie ze schematem przedstawionym na rysunku 3.1. W celu estymacji jakości syntezowano widok wirtualny w pozycji widoku rzeczywistego, umożliwiając porównanie zsyntezowanego widoku z widokiem odniesienia.

3.2. Wybór metody odniesienia

W celu oceny opracowanej przez autora metody syntezy widoków wirtualnych MVS, jakość widoków syntezowanych przy użyciu metody MVS została porównana z jakością widoków wirtualnych syntezowanych przy użyciu oprogramowania odniesienia rozwijanego przez grupę ekspertów MPEG (ISO/IEC JTC 1/SC29/WG 11) – VSRS (*View Synthesis Reference Software*) w najnowszej wersji dostępnej w trakcie realizacji pracy – wersji 4.2 [Senoh'17A].

Podstawową zaletą oprogramowania odniesienia VSRS jest fakt, iż jego kod źródłowy jest dostępny dla wszystkich naukowców prowadzących badania w dziedzinie przetwarzania sekwencji wielowidokowych. Z tego powodu, w większości prac badawczych dotyczących syntezy widoków wirtualnych, autorzy porównują zaproponowane przez siebie metody z oprogramowaniem VSRS [Ceulemans'18, Dziembowski'16A, Fachada'18, Jin'16, Jorissen'14, Jun-Te'15, Li'18, Zhu'16B].

Ostatnim powodem, dla którego zdecydowano się porównać opracowaną metodę syntezy z VSRS jest brak ograniczeń co do charakterystyki systemu wielokamerowego. W odróżnieniu od metod wymagających określonego rozmieszczenia kamer (np. [Akin'15, Jun-Te'15, Tanimoto'08A]), VSRS umożliwia poprawną syntezę widoku wirtualnego dla dowolnie rozstawionych kamer.

3.3. SPOSÓB PRZEPROWADZANIA EKSPERYMENTÓW

Opracowana przez autora i opisana w dalszej części rozprawy metoda syntezy MVS zawiera szereg technik, których użycie może być osobno włączane lub wyłączane. Aby wyznaczyć poprawę jakości spowodowaną zastosowaniem różnych zaproponowanych technik poprawiających jakość syntezy, zbadano wpływ każdej techniki z osobna. W tym celu dla każdej techniki przeprowadzano dodatkowy eksperyment, w którym technika ta nie była używana.

W celu subiektywnego zmierzenia jakości zastosowano metodę PC (*Pair Comparison*) rekomendowaną przez ITU-T [ITU-T P.910]. Wybór tej metody uzasadniono w punkcie 3.6.2. Testy subiektywne przeprowadzono dla czterech scenariuszy:

- 1. Porównanie opracowanej metody z oprogramowaniem odniesienia VSRS: rozdział 10.2.
- 2. Wpływ korekcji niespójności barwnej (włączona/wyłączona korekcja niespójności barwnej): rozdział 8.4.
- 3. Wpływ filtracji (włączona/wyłączona filtracja wirtualnej mapy glębi i samego widoku wirtualnego): rozdział 6.3.

4. Wpływ rzutowania zapewniającego ciągłość obiektów (przypadek wirtualnego zbliżenia się do zarejestrowanych obiektów): rozdział 7.3.

Przeprowadzenie metodologicznie poprawnych testów subiektywnych wymaga udziału wielu uczestników. Ponadto, aby zmęczenie uczestników nie wypaczyło wyników, sesja testowa nie może trwać zbyt długo [ITU-R BT.500-9]. Czas trwania sesji testowej wyniósł około 30 minut, a więc nie przekroczono sugerowanego w normie czasu testów.

3.4. WYKORZYSTANE SEKWENCJE TESTOWE

Wykorzystany zbiór testowy składał się z 12 zróżnicowanych sekwencji wielowidokowych: BBB Butterfly, BBB Flowers [Kovacs'15B], Poznan_Blocks2, Poznan_Fencing2, Poznan_Service2 [Domański'16F], Ballet, Breakdancers [Zitnick'04], Poznan_Carpark, Poznan_Street [Domański'09B], Soccer Linear [Goorts'14B], Soccer Arc [Goorts'14C] i Poznan_Blocks [Domański'14C].

Wybrane sekwencje testowe są powszechnie używane, zarówno przez niezależnych badaczy (np. [Ceulemans'18, Li'18, Luo'17, Mieloch'17A, Schmeing'15, Tezuka'15]), jak i grupę ekspertów MPEG [Teratani'18].

Na rysunku 3.2 przedstawiono przykładowe ramki sekwencji testowych, zarejestrowane przez centralną kamerę systemu wielokamerowego (kamerę 45. dla sekwencji *BBB Butterfly* i *BBB Flowers* oraz kamerę 4. dla pozostałych sekwencji).

W zbiorze umieszczono 2 sekwencje syntetyczne (*BBB Butterfly*, *BBB Flowers*) wygenerowane komputerowo (za pomocą oprogramowania Blender) i posiadające perfekcyjne mapy glębi, a także 10 sekwencji naturalnych, zarejestrowanych rzeczywistymi systemami wielokamerowymi. Sekwencje dobrano tak, by jakość map glębi wyznaczonych dla poszczególnych sekwencji była zróżnicowana. Przykładowo mapy glębi dla sekwencji *Poznan_Service2* i *Poznan_Blocks* [Wegner'14] charakteryzują się rozmytymi krawędziami, a wartości glębi dla sekwencji *Soccer Arr* są w niepoprawny sposób przeskalowane, co skutkuje nieprecyzyjnym rzutowaniem obiektów. Zdecydowanie lepszą jakość mają mapy glębi wyznaczone dla sekwencji *Ballet* i *Breakdancers*, a także *Poznan_Fencing2* i *Poznan_Blocks2*, wyznaczone przy użyciu algorytmów bazujących na segmentacji (odpowiednio [Zitnick'04] i [Mieloch'17A]).

Wszystkie sekwencje zarejestrowane zostały przy użyciu co najmniej siedmiu kamer. Dla obu sekwencji syntetycznych dostępnych było 79 widoków, spośród których do większości testów wybrano 7 równomiernie rozmieszczonych widoków rzeczywistych.

Długość większości sekwencji testowych wynosi 10 sekund. Wyjątkiem są obie sekwencje syntetyczne *BBB*, dla których 120 dostępnych ramek przy 24 obrazach na sekundę przekłada się na 5 sekund długości sekwencji, a także *Ballet* i *Breakdancers*, dla których długość wyniosła 6,7 sekundy.

Metodologia badań



Poznan_Carpark Rysunek 3.2. Przykładowe ramki sekwencji testowych

W zbiorze sekwencji testowych umieszczono 9 sekwencji zarejestrowanych przy użyciu rzadkich systemów wielokamerowych i 3 sekwencje nagrane systemami z gęstym, liniowym rozmieszczeniem kamer (*Poznan_Carpark, Poznan_Street, Soccer Linear*). Pozostałe sekwencje zostały zarejestrowane systemami z kamerami umieszczonymi na łuku, z kątem pomiędzy osiami optycznymi skrajnych kamer wynoszącym od 30 stopni (*Ballet, Breakdancers*) do 100 stopni (*Poznan_Blocks*). Dla większości sekwencji testowych kamery rozmieszczone były w przybliżeniu równomiernie, dla trzech z nich – parami (*Poznan_Blocks2, Poznan_Fencing2, Poznan_Service2*).

Użyte sekwencje charakteryzują się różnymi rozmiarami obrazu, od 1024×768 (Ballet, Breakdancers), poprzez 1280×768 (BBB Butterfly, BBB Flowers) i 1392×1136 (Soccer Linear) aż do 1920×1080 dla pozostałych sekwencji.

Sekwencja testowa	BBB Butterfly	BBB Flowers	Poznan_Blocks	Poznan_Blocks2	Poznan_Fencing2	Poznan_Service2	Ballet	Breakdancers	Soccer Arc	Soccer Linear	Poznan_Carpark	Poznan_Street
Rodzaj sekwencji naturalna (N)/syntetyczna (S)	SA	SA	Ν	Ν	Ν	Ν	Ν	Ν	Ν	Ν	Ν	Ν
Ustawienie kamer liniowe (1)/łukowe (2)	2	2	2	2	2	2	2	2	2	1	1	1
Kąt między osiami optycznymi skrajnych kamer	39.5°	39.5°	100°	60°	60°	60°	30°	30°	60°	_	-	-
Liczba widoków rzeczywistych	7 (79 ^B)	7 (79 ^B)	10	10	10	10	8	8	7	7	7	7
Ustawienie kamer równomierne (R)/parami (P) ^C	R	R	R	Р	Р	Р	R	R	R	R	R	R
Rozmiar obrazu (liczba kolumn/linii)	1280 768	1280 768	1920 1080	1920 1080	1920 1080	1920 1080	1024 768	1024 768	1920 1080	1392 1136	1920 1088	1920 1088
Użyte do syntezy widoki rzeczywiste	6 – 84 (co 13 ^D)	6 – 84 (co 13 ^D)	0 – 9	0 – 9	0 – 9	0 – 9	0-7	0-7	1 – 7	1 – 7	1 – 7	1 – 7
Widok rzeczywisty, dla którego wyznaczano jakość subiektywną	45	45	4	4	4	4	4	4	4	4	4	4
Długość sekwencji (w sekundach)	5 ^E	5 ^E	10	10	10	10	6.7 ^F	6.7 ^F	10	10	10	10
Liczba ramek w sekwencji	120	120	250	250	250	250	100	100	250	600	250	250
Liczba ramek na sekundę ^G	24	24	25	25	25	25	15	15	25	60	25	25

Tabela 3.1. Charakterystyka sekwencji testowych

^A – Sekwencja wraz z mapami głębi wygenerowana w programie Blender [Kovacs'15B].

^B – Dostępne jest 79 widoków, do większości testów użyto 7 widoków rzeczywistych.

^C – Kamery systemu rozstawione w stereoparach w celu lepszej estymacji glębi [Domański'16G].

^D – Wybór co trzynastego widoku zgodny z wytycznymi grupy MPEG [Lafruit'15B].

^E – Podczas testów subiektywnych prezentowano dwukrotnie wydłużoną sekwencję; ostatnie 120 ramek prezentowano w odwrotnej kolejności w celu uniknięcia nieciągłości ruchu.

F – Podczas testów subiektywnych prezentowano półtorakrotnie wydłużoną sekwencję; ostatnie 50 ramek prezentowano w odwrotnej kolejności w celu uniknięcia nieciągłości ruchu.

^G – Podczas testów subiektywnych wszystkie sekwencje prezentowano w odpowiednim tempie, prezentując kolejne ramki z taką częstotliwością, z jaką zarejestrowano poszczególne sekwencje.

3.5. INNE ZBIORY MATERIAŁÓW TESTOWYCH

W przypadku badań dotyczących zagadnień pokrewnych do opisywanej w rozprawie syntezy widoków wirtualnych, najczęściej wykorzystywanym zbiorem materiałów testowych jest baza sekwencji testowych Middlebury [Middlebury]. Zawiera ona klika zbiorów statycznych obrazów

Metodologia badań

testowych wraz z mapami głębi: [Hirschmüller'07, Scharstein'02, Scharstein'03, Scharstein'14]. Niewątpliwą zaletą bazy Middlebury jest znaczna liczba obrazów testowych (łącznie ponad 70), dostępność map głębi, a także możliwość porównania własnej metody z kilkudziesięcioma algorytmami przetestowanymi na sekwencjach z tej bazy.

Niestety, porównanie zaproponowanej metody z licznymi algorytmami opracowanymi przez zespoły badawcze z całego świata jest możliwe **wyłącznie w przypadku metod estymacji głębi**. Tym samym, dla syntezy widoków wirtualnych nie jest możliwe wykorzystanie podstawowej zalety bazy Middlebury.

Baza Middlebury mogłaby służyć również jako dodatkowy zbiór obrazów testowych, przy użyciu którego opracowana metoda MVS mogłaby zostać porównana z algorytmem odniesienia VSRS. Jednakże, pomimo iż dla większości zarejestrowanych sceny dostępnych jest kilka widoków (7 - 9), odpowiadające im **mapy głębi dostępne są wyłącznie dla 2 widoków**. Tym samym, do syntezy widoku wirtualnego można by wykorzystać wyłącznie 2 widoki rzeczywiste. Podstawową cechą opracowanej metody jest natomiast wykorzystanie większej liczby widoków.

Ponadto, celem rozprawy było opracowanie metody syntezy widoków wirtualnych pozwalającej osiągnąć dobrą jakość syntezowanych widoków dla systemów wielokamerowych z rzadkim rozmieszczeniem kamer, w tym dla systemów z dowolnie rozmieszczonymi kamerami. Poszczególne widoki z zestawów obrazów w bazie Middlebury zostały zaś zarejestrowane za pomocą kamer **stojących w linii** i znajdujących się w **bliskiej odległości** od siebie [Middlebury, Scharstein'02, Scharstein'14].

Biorąc pod uwagę wymienione wady i niedoskonałości bazy Middlebury dla zastosowań syntezy widoków wirtualnych w rzadkich systemach wielokamerowych, obrazy testowe z tej bazy nie zostały wykorzystane w badaniach opisanych w rozprawie.

3.6. POMIAR JAKOŚCI SYNTEZOWANYCH WIDOKÓW WIRTUALNYCH

3.6.1. OBIEKTYWNY POMIAR JAKOŚCI

3.6.1.1. PSNR

Pierwszą z zastosowanych metryk jest PSNR (*Peak Signal-to-Noise Ratio*). Dla obrazu monochromatycznego jest on wyrażony wzorem:

$$PSNR[dB] = 10 \log_{10}\left(\frac{(2^n - 1)^2}{MSE}\right),$$
 (3.1)

gdzie:

-n jest liczbą bitów na punkt obrazu (w opisywanych w rozprawie przypadkach n = 8),

- MSE wyrażany jest jako:

$$MSE = \frac{1}{W \cdot H} \sum_{y=0}^{H-1} \sum_{x=0}^{W-1} \left(S(x, y) - R(x, y) \right)^2, \qquad (3.2)$$

gdzie:

- W i H określają liczbę próbek odpowiednio w jednym wierszu i jednej kolumnie obrazu,
- S(x, y) i R(x, y) są wartościami próbek, odpowiednio, obrazu syntezowanego i obrazu odniesienia.

W rozprawie przyjęto, że PSNR będzie liczony wyłącznie dla składowej Y obrazu. Wyznaczanie jakości obrazu wyłącznie dla luminancji jest powszechną, często stosowaną w literaturze praktyką (np. [Ceulemans'18, Dziembowski'16A, Fachada'18A, Jun-Te'15, Li'18, Schmeing'15]).

W przypadku syntezowanych widoków wirtualnych, miara PSNR nie zawsze odzwierciedla jakość mierzoną w sposób subiektywny. Dla przykładu, często występujące w syntezowanym widoku minimalne przesunięcie krawędzi obiektu (np. o 1 okres próbkowania) [Purica'15] nie zmienia wrażenia widzów, a redukuje jakość widoku wirtualnego mierzoną poprzez PSNR. Innym przykładem mogą być blędy wynikające z braku spójności czasowej [Mieloch'17B], widoczne dla użytkownika jako niewielkie migające obiekty czy też falujące krawędzie obiektów większych. Z powodu niewielkiego rozmiaru zmniejszają one wartość PSNR nieznacznie, jednocześnie skupiając na sobie uwagę użytkownika, a więc mocniej degradując jakość subiektywną.

Pomimo wymienionych wad, PSNR cechuje się prostotą, przez co w większości publikacji z dziedziny syntezy widoków wirtualnych jakość syntezowanych widoków wyznaczana jest właśnie poprzez PSNR.

3.6.1.2. SSIM

Drugą użytą w rozprawie miarą jakości obiektywnej jest miara podobieństwa strukturalnego (ang. *Structural Similarity*) – SSIM [Wang'04]. Metryka ta została stworzona, by lepiej zasymulować system widzenia człowieka, a tym samym bardziej odzwierciedlać wyniki testów subiektywnych. W odróżnieniu od PSNR, gdzie różnica pomiędzy dwoma obrazami liczona jest niezależnie dla każdego punktu obrazu, w SSIM uwzględniane są lokalne charakterystyki obrazu, estymowane wewnątrz przesuwanego po obrazach okna:

$$SSIM(s,r) = \frac{(2 \cdot \mu_s \cdot \mu_r + C_1) \cdot (2 \cdot \sigma_{sr} + C_2)}{(\mu_s^2 + \mu_r^2 + C_1) \cdot (\sigma_s^2 + \sigma_r^2 + C_2)},$$
(3.3)

gdzie:

Metodologia badań

- s oraz r są kwadratowymi, odpowiadającymi sobie obszarami, odpowiednio, widoku syntezowanego i odniesienia (w pracy przyjęto typowy rozmiar okna równy 11×11 punktów; za pracami: [Golestani'14, Malpica'09, Wang'04]).
- μ_s i μ_r wartość oczekiwana luminancji w obszarach s i r,
- σ_s^2 i σ_r^2 wariancja luminancji w obszarach *s* i *r*,
- σ_{sr} kowariancja luminancji w obszarach *s* i *r*,
- C_1 i C_2 wartości stałe uniemożliwiające dzielenie przez 0; zgodnie z [Wang'04] wynoszące odpowiednio $(0,01 \cdot 255)^2$ i $(0,03 \cdot 255)^2$.

Jakość dla całego obrazu mierzona jest jako średnia wartość SSIM(s, r) dla wszystkich położeń analizowanego okna.

Pomimo większego skomplikowania i dłuższego czasu estymacji jakości, w przypadku syntezy widoków wirtualnych również miara SSIM nie zawsze odpowiada wynikom testów subiektywnych [Jang'15]. Niemniej jednak, jest ona często wykorzystywaną metodą estymacji jakości syntezowanych widoków, dlatego też została ona uwzględniona w rozprawie.

3.6.2. SUBIEKTYWNY POMIAR JAKOŚCI

Do subiektywnego pomiaru jakości użyto metody porównywania dwóch obrazów – PC (*Pair Comparison*) rekomendowanej przez ITU-T [ITU-T P.910].

Zgodnie z [vanDijk'97] i [Mantiuk'12], metoda porównywania dwóch obrazów (2 sekwencji) jest lepsza od ich indywidualnego oceniania w przypadku, gdy znacząco różni się charakter występujących w nich blędów. W przypadku wybranego zbioru testowego treść sekwencji, ich stopień skomplikowania, a przez to i charakter pojawiających się w widokach wirtualnych blędów są bardzo zróżnicowane. Również w pracy [Lee'13B] wykazano, iż metoda PC jest odpowiednią metodą do oceny jakości subiektywnej syntezowanych widoków.

Przy przygotowaniu i przeprowadzaniu testów kierowano się wytycznymi zawartymi w normach ITU-T Rec. P.910 [ITU-T P.910] i ITU-R Rec. BT.500-9 [ITU-R BT.500-9]. Udział w testach wzięło 44 uczestników. Czas trwania jednej sesji testowej wyniósł 30 minut. Każdy z uczestników przeprowadzał 96 testów (4 scenariusze, 12 sekwencji testowych i 2 powtórzenia dla każdej sekwencji, w tym jedno ze zmienionym uporządkowaniem wyświetlania).

Przez pierwsze 10 sekund każdego testu uczestnik oglądał parę umieszczonych obok siebie sekwencji. Z jednej strony umieszczona była sekwencja zsyntezowana za pomocą testowanego algorytmu a, z drugiej – w celu porównania – ta sama sekwencja zsyntezowana za pomocą algorytmu b. Przykładowo, przy porównywaniu opracowanej w ramach rozprawy metody MVS z metodą odniesienia VSRS, z jednej strony umieszczona była sekwencja zsyntezowana za pomocą MVS,

z drugiej – za pomocą VSRS. Analogicznie, w przypadku badania wpływu korekcji niespójności barwnej, z jednej strony umieszczana była sekwencja zsyntezowana z użyciem korekcji niespójności barwnej, z drugiej zaś ta sama sekwencja zsyntezowana bez korekcji.

Po trwającym 10 sekund oglądaniu pary sekwencji, uczestnik miał do 10 sekund na dokonanie oceny.

Sesja testowa była przeprowadzana w sposób przedstawiony na rysunku 3.3.



Rysunek 3.3. Sposób przeprowadzania sesji testowej (na podstawie [ITU-T P.910]). Dużymi literami (A, B, C) oznaczono różne sekwencje testowe, a oraz b oznaczają algorytmy użyte do syntezy widoków wirtualnych. Zapis A(a, b) oznacza, iż po lewej stronie ekranu prezentowana była wirtualna sekwencja widoków zsyntezowana dla sekwencji testowej A przy użyciu algorytmu a, po prawej – ta sama sekwencja zsyntezowana przy użyciu algorytmu b

Dla każdego testowanego scenariusza uczestnik porównywał jakość dla każdej sekwencji testowej dla obu możliwych uporządkowań porównywanych algorytmów syntezy, a więc (a, b) i (b, a). Kolejność prezentacji poszczególnych par sekwencji była niezależnie losowana dla każdego z uczestników, a więc jeden z uczestników mógł oceniać pary sekwencji w kolejności A(a, b), B(b, a), C(a, b) itd., podczas gdy drugi dokonywał oceny dla kolejności B(a, b), C(a, b), B(b, a), A(b, a) itd.

Większość sekwencji testowych wchodzących w skład przyjętego zbioru testowego charakteryzuje się wysoką rozdzielczością (FullHD – 1920×1080). Aby nie zakłamywać wyników poprzez konieczność zmniejszania rozdzielczości syntezowanych sekwencji, testy subiektywne przeprowadzono z użyciem dwóch identycznych monitorów FullHD. Sekwencje niższej rozdzielczości (niewypełniające całej powierzchni roboczej monitorów) były wyśrodkowane i otoczone jednolitym, szarym tłem.

W fazie oceny uczestnicy oceniali, która z dwóch prezentowanych sekwencji ma lepszą jakość, używając do tego zalecanej 7-stopniowej skali, od -3 do 3. W odróżnieniu od wariantu metody PC z wymuszonym wyborem, uczestnik mógł wybrać 0, jeżeli dla danej pary nie dostrzegł różnicy jakości.

Zebrane wyniki przetworzono zgodnie z wytycznymi [ITU-R BT.500-9] i zaprezentowano w dalszej części rozprawy.

Metodologia badań

4. SYNTEZA WIDOKÓW WIRTUALNYCH W RZADKICH SYSTEMACH WIELOKAMEROWYCH: ZAŁOŻENIA I WYMAGANIA

W rozdziale 4. dokonano analizy trzech przedstawionych w rozdziale 1.3 problemów występujących w systemach z rzadkim rozmieszczeniem kamer: niewielkiego obszaru sceny widocznego jednocześnie w różnych widokach rzeczywistych, ograniczonej rozdzielczości obrazu i nielambertowskich odbić światła. W dalszych punktach szerzej omówiono powyższe problemy, przedstawiając sposób ich rozwiązania. Sposób ten – wypełnianie odsłonięć przy użyciu informacji z dalszych widoków rzeczywistych – stał się podstawą opracowanej przez autora i opisanej w dalszej części rozprawy metody syntezy widoków wirtualnych (MVS).

4.1. Odsłonięcia

Pierwszym z analizowanych problemów charakterystycznych dla rzadkich systemów wielokamerowych są odsłonięcia, a więc te obszary widoku wirtualnego, które w widokach rzeczywistych były przysłonięte przez obiekty znajdujące się bliżej kamer.

Dla dowolnej pary widoków rzeczywistych możliwe jest wyznaczenie obszaru wspólnego, zawierającego wszystkie punkty sceny widoczne jednocześnie w obu widokach. Ideę estymacji obszaru wspólnego pokazano na rysunku 4.1.



Rysunek 4.1. Wyznaczanie obszaru wspólnego dla dwóch sąsiednich widoków rzeczywistych; kolorem czarnym oznaczono obszar "nie-wspólny" – te fragmenty widoku, które nie były widoczne w widoku sąsiednim (dla czytelności rysunku nie uwzględniono wyznaczania obszaru wspólnego dla białego tła; w rzeczywistym przypadku powierzchnia obszaru wspólnego jest wyznaczana również dla tła)

Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagania

W górnym wierszu przedstawiono dwa sąsiednie widoki rzeczywiste, i oraz (i + 1). W wierszu dolnym pokazano obszar wspólny dla obu przedstawionych widoków – jest to cały obszar obrazu poza fragmentami zaznaczonymi na czarno, które to fragmenty stanowią te części widoku, które nie są widoczne w widoku sąsiednim. Jak pokazano, obszary te są różne dla obu widoków, a więc i wielkość obszaru wspólnego dla widoków (i + 1) oraz i może być różna w zależności od tego, w którym z dwóch widoków jest ona liczona.

Rzadki system wielokamerowy cechuje się dużymi odległościami między kamerami lub dużym kątem pomiędzy osiami optycznymi poszczególnych kamer. Obie wymienione cechy mają ten sam skutek – stosunkowo niewielka wspólna część sceny jest rejestrowana przez dowolne dwie sąsiednie kamery systemu. Oczywiście, wielkość obszaru rejestrowanego przez sąsiednie kamery może sięgać nawet, według badań przeprowadzonych przez autora na zbiorze zróżnicowanych sekwencji testowych (tabela 4.1), 90% całego obrazu. Dla porównania jednak, w sekwencjach zarejestrowanych z użyciem prostych systemów z liniowym ustawieniem kamer [Domański'09B, Goorts'14B] ten udział może sięgać ponad 95% (tabela 4.1).

	Sekwencja	Obszar wspólny [%]			
Rzadki system wielokamerowy; kamery ustawione na łuku	BBB Butterfly	67,03			
	BBB Flowers	66,18			
	Poznan_Blocks	69,87			
	Poznan_Blocks2	81,01			
	Poznan_Fencing2	83,47			
	Poznan_Service2	75,21			
	Ballet	84,00			
	Breakdancers	90,62			
	Soccer Arc	76,68			
Kamery ustawione liniowo	Soccer Linear	95,38			
	Poznan_Carpark	94,60			
	Poznan_Street	96,12			

Tabela 4.1. Wspólny obszar sceny rejestrowany przez dwie sąsiednie kamery systemu (wartość uśredniona dla wszystkich par sąsiednich kamer; wartości wyznaczone przez autora rozprawy)

W celu wyznaczenia obszaru wspólnego dla zbioru sekwencji testowych, dla każdej pary sąsiednich widoków rzeczywistych przerzutowano (zgodnie z (1.3)) wszystkie punkty jednego widoku do widoku drugiego, a następnie zliczono wszystkie te punkty drugiego widoku, do których nie został przerzutowany żaden punkt z widoku pierwszego. Następnie tę samą operację wykonywano w drugą stronę, rzutując punkty drugiego widoku do widoku pierwszego. Wartości zamieszczone w tabeli 4.1

wyznaczono poprzez uśrednienie wielkości wspólnego obszaru dla wszystkich par sąsiadujących widoków rzeczywistych.

Zbiór sekwencji testowych w tabeli 4.1 podzielono na dwa podzbiory. W pierwszym z nich znajdują się sekwencje zarejestrowane przy użyciu rzadkich systemów wielokamerowych z kamerami rozmieszczonymi na łuku. Średnia wielkość wspólnego obszaru sceny rejestrowanego przez dowolne sąsiednie kamery systemu dla tych sekwencji wyniosła 77% i jest zdecydowanie mniejsza, niż w przypadku sekwencji zarejestrowanych systemami z liniowym rozmieszczeniem kamer, dla których średnia wielkość obszaru wspólnego wyniosła ponad 95%.

W przypadku próby syntezy widoku wirtualnego na podstawie wyłącznie jednego widoku rzeczywistego, cały pozostały ("nie-wspólny") obszar obrazu (a więc średnio 23% dla sekwencji zarejestrowanych systemami z kamerami rozmieszczonymi na łuku) musiałby zostać uzupełniony (interpolowany bądź ekstrapolowany) na podstawie informacji z najbliższych przerzutowanych punktów.

W typowym podejściu, synteza widoków wirtualnych przeprowadzana jest jednak przy użyciu dwóch sąsiednich widoków rzeczywistych. W takim przypadku uzupełnione muszą zostać jedynie obszary niewidoczne w żadnym z sąsiednich widoków, co zostało zobrazowane na rysunku 4.2.



Rysunek 4.2. Wyznaczanie obszarów widocznych w co najmniej jednym sąsiednim widoku rzeczywistym; kolorem czarnym oznaczono obszar "nie-wspólny" – te fragmenty widoku, które nie były widoczne w żadnym sąsiednim widoku (dla czytelności rysunku nie uwzględniono wyznaczania obszaru wspólnego dla białego tła; w rzeczywistym przypadku powierzchnia obszaru wspólnego jest wyznaczana również dla tła)

W tabeli 4.2 pokazano jaka część widoku *i* jest widoczna również w widoku (i - 1) lub (i + 1) dla używanych sekwencji testowych. Dla każdej sekwencji w tabeli podano wielkość uśrednioną dla wszystkich widoków rzeczywistych mających 2 sąsiednie widoki rzeczywiste (a więc pominięto skrajnie lewy i skrajnie prawy widok rzeczywisty).

Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagania

	Sekwencja	Obszar widoczny w widokach sąsiednich [%]			
y;	BBB Butterfly	87,34			
Rzadki system wielokamerowy kamery ustawione na łuku	BBB Flowers	86,27			
	Poznan_Blocks	88,64			
	Poznan_Blocks2	95,22			
	Poznan_Fencing2	95,81			
	Poznan_Service2	89,71			
	Ballet	95,01			
	Breakdancers	95,73			
	Soccer Arc	87,50			
umery awione dowo	Soccer Linear	98,46			
	Poznan_Carpark	98,39			
Ká usta lin	Poznan_Street	98,93			

Tabela 4.2. Wielkość obszaru widocznego w co najmniej jednym sąsiednim widoku rzeczywistym (wartość uśredniona dla wszystkich widoków rzeczywistych posiadających widoki sąsiednie; wartości wyznaczone przez autora rozprawy)

Jak pokazano w tabeli 4.2, obszar widoczny w którymkolwiek z sąsiednich widoków rzeczywistych jest znacząco większy dla sekwencji zarejestrowanych przy użyciu gęstych systemów wielokamerowych z liniowym rozmieszczeniem kamer. Dla trzech sekwencji testowych zarejestrowanych przy pomocy takiego systemu, powierzchnia odsłonięć, a więc obszarów niewidocznych w żadnym z widoków rzeczywistych wynosi średnio około 1,4% powierzchni całego obrazu. W przypadku sekwencji rzadkich, średnia ta wynosi 9% powierzchni obrazu, a więc sześciokrotnie więcej.

Powierzchnia odsłonięć waha się w zależności od charakterystyki sceny i samego rozmieszczenia kamer. Dla przykładu powierzchnia odsłonięć dla sekwencji *BBB Flowers* (w której jest wiele wzajemnie przysłaniających się obiektów) jest większa, niż dla sekwencji *Ballet* (gdzie w scenie oprócz tła są tylko dwie osoby). Zależność powierzchni odsłonięć od rozmieszczenia kamer obrazuje natomiast para sekwencji *Poznan_Blocks* i *Poznan_Blocks2*. Pomimo bardzo podobnej charakterystyki (dwie osoby układające klocki na stole), w sekwencji *Poznan_Blocks2* powierzchnia odsłonięć jest znacząco mniejsza, co spowodowane jest ustawieniem kamer w parach. Powodem takiego stanu rzeczy jest fakt, iż kamery tworzące dowolną stereoparę rejestrują scenę z podobnego punktu widzenia, co przekłada się na duży wspólny obszar w obu widokach.

Niemniej jednak, zaprezentowane wyniki wyraźnie sugerują, iż rozwiązanie problemu dużej powierzchni odsłonięć w sekwencjach zarejestrowanych przy użyciu rzadkich systemów wielokamerowych umożliwi uzyskanie lepszej jakości widoków wirtualnych. Rozwiązaniem tego problemu jest wykorzystanie informacji z większej liczby widoków rzeczywistych niż tylko dwa sąsiednie. Zaprezentowane podejście pozornie wydaje się doskonałym rozwiązaniem, jednakże należy zwrócić uwagę na dwa pozostałe problemy występujące w rzadkich systemach wielokamerowych, opisane w punktach 4.2.1 i 4.2.2.

4.2. ZALETY SYNTEZY Z NAJBLIŻSZYCH WIDOKÓW RZECZYWISTYCH

4.2.1. PROBLEM OGRANICZONEJ ROZDZIELCZOŚCI OBRAZU

Drugim problemem, jaki należy rozwiązać jest ograniczona rozdzielczość obrazu. Problem ten zostanie omówiony na przykładzie przedstawionym na rysunkach 4.3 i 4.4.



Rysunek 4.3. Problem ograniczonej rozdzielczości obrazu w syntezie widoków wirtualnych: prosta scena, widok z góry

Na rysunku 4.3 pokazano widok z góry na przykładową, prostą scenę, w której znajduje się jeden płaski obiekt stojący na środku sceny. Scena jest rejestrowana przez 4 kamery rzeczywiste oznaczone od (i + 1) do (i + 4). W omawianym przykładzie syntezowany będzie widok rejestrowany przez wirtualną kamerę (i + 2,5). Widok ten syntezowany będzie czterokrotnie, za każdym razem przy użyciu innego widoku rzeczywistego (rys. 4.4).



Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagania

Rysunek 4.4. Problem ograniczonej rozdzielczości obrazu w syntezie widoków wirtualnych: rzutowanie tego samego obiektu do widoku wirtualnego (i + 2,5) z różnych widoków rzeczywistych

Rysunek 4.4 przedstawia rzutowanie (zgodnie z (1.3)) obiektu z prostej sceny przedstawionej na rysunku 4.3 z poszczególnych widoków rzeczywistych do widoku wirtualnego (i + 2.5). W przypadku rzutowania obiektu z widoków zarejestrowanych przez kamery (i + 1) oraz (i + 2)ograniczona rozdzielczość obrazu nie wpływa na jakość syntezowanego widoku – kąt pomiędzy długim bokiem obiektu a osiami optycznymi obu wspomnianych kamer jest bowiem bliższy 90 stopniom, niż w przypadku osi optycznej kamery (i + 2,5). Osie optyczne kamer (i + 3) oraz (i + 4) są skierowane pod zdecydowanie większym kątem względem długiej ściany obiektu, przez co po przerzutowaniu do widoku (i + 2,5) w obiekcie pojawiły się pęknięcia (ang. *cracks*) – tym większe, im z dalszego widoku obiekt został przerzutowany.

Jak pokazano na rysunku 4.4, występowanie pęknięć spowodowanych ograniczoną rozdzielczością obrazu ściśle związane jest z kątem, pod jakim obserwowany jest obiekt.

4.2.2. PROBLEM NIELAMBERTOWSKICH ODBIĆ ŚWIATŁA

Wydawać by się mogło, iż w przypadku pokazanym na rysunkach 4.3 i 4.4 do syntezy widoku (i + 2,5) lepiej jest – oprócz lewego sąsiedniego widoku (i + 2) – użyć kolejnego lewego widoku (i + 1), niż sąsiedniego prawego (i + 3). Niestety, podczas syntezy widoków wirtualnych w rzadkich systemach wielokamerowych należy wziąć pod uwagę jeszcze jeden problem, a mianowicie różne oświetlenie sceny w poszczególnych widokach. Obiekty znajdujące się w rzeczywistej scenie rejestrowanej przez system wielokamerowy nie mają idealnie rozpraszającej, lambertowskiej powierzchni [Koppal'14]. Rzeczywiste, nielambertowskie matowe powierzchnie nie rozpraszają światła w sposób izotropowy, ale zależny od kierunku padania światła [Tagare'91, Zhou'13]. Tym

samym jasność obiektu będzie różnie rejestrowana przez poszczególne kamery systemu wielokamerowego (rysunki 4.5 i 4.6).



Rysunek 4.5. Problem nielambertowskich odbić w syntezie widoków wirtualnych: prosta scena, widok z góry; długość strzałek obrazuje natężenie światla odbitego od obiektu

Na rysunku 4.5 pokazano tę samą prostą scenę, co na rysunku 4.3, przy czym w scenie tej umieszczone zostało punktowe źródło światła. Stojący w środku sceny obiekt ma matową, acz nielambertowską powierzchnię, przez co światło odbite od obiektu w kierunku różnych kamer ma różne natężenie. Obraz zarejestrowany przez poszczególne kamery pokazano na rysunku 4.6.



Rysunek 4.6. Problem nielambertowskich odbić w syntezie widoków wirtualnych: rzutowanie obiektu do widoku wirtualnego (*i* + 2,5) z różnych widoków rzeczywistych; obiekt rejestrowany przez kamery (*i* + 1) oraz (*i* + 4) jest ciemniejszy, niż dla kamer (*i* + 2) oraz (*i* + 3)

W górnym rzędzie rysunku 4.6 pokazano obrazy zarejestrowane przez rzeczywiste kamery systemu, w rzędzie dolnym – obrazy przerzutowane do pozycji wirtualnej kamery (i + 2,5). Zmiana natężenia światła docierającego do poszczególnych kamer jest tym większa, im większa jest odległość (w tym przypadku kątowa) pomiędzy nimi, a więc jasność obiektu zsyntezowanego na podstawie

Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagania

sąsiednich widoków (i + 2) oraz (i + 3) będzie zdecydowanie bliższa prawdziwej jasności obiektu obserwowanego z perspektywy kamery (i + 2,5) niż w przypadku syntezy z widoków dalszych: (i + 1) czy (i + 4).

Zaprezentowany na rysunkach 4.5 i 4.6 przykład prezentuje zjawisko nielambertowskich odbić światła w sposób uproszczony. W rzeczywistym przypadku powierzchnia obiektu może odbijać światło różnie dla różnych długości fali świetlnej, przez co ten sam obiekt zarejestrowany przez różne kamery rzeczywiste może mieć nie tylko różną jasność, ale i odcień.

4.2.3. EKSPERYMENT – SYNTEZA Z UŻYCIEM RÓŻNYCH WIDOKÓW RZECZYWISTYCH

Aby potwierdzić słuszność przeprowadzanych w punktach 4.2.1 i 4.2.2 rozważań dla użytego zbioru sekwencji testowych, przeprowadzono eksperyment sprawdzający, jak wybór widoków rzeczywistych używanych do syntezy widoku wirtualnego wpływa na jakość widoku wirtualnego. Dla każdej sekwencji jako widok odniesienia wybrano środkowy widok rzeczywisty (dla obu sekwencji *BBB* jest to widok 45, dla pozostałych sekwencji widok 4). Następnie syntezowano widok wirtualny w pozycji widoku odniesienia, w celu wyznaczenia jakości syntezowanego widoku mierzonej za pomocą miary PSNR. Za każdym razem do syntezy wykorzystywano dwa widoki rzeczywiste, przy czym w kolejnych iteracjach wybierano inne widoki w celu przetestowania wszystkich możliwych kombinacji. Syntezę przeprowadzano przy użyciu oprogramowania odniesienia – VSRS [Senoh'17A].

Sposób przeprowadzenia eksperymentu można przedstawić na przykładzie hipotetycznej sekwencji zarejestrowanej przez 5 kamer (ponumerowanych kolejno 0, 1, 2, 3 i 4). W takim przypadku widokiem odniesienia wybrano widok 2 (widok z centralnej kamery systemu), a pozostałe widoki rzeczywiste były używane do syntezy widoku wirtualnego w pozycji widoku 2. Syntezy widoku wirtualnego dokonywano przy użyciu wszystkich możliwych par widoków rzeczywistych: 00, 01, 03, 04, 10, 11, 13, 14, 30, 31, 33, 34, 40, 41, 43 i 44. W ostatnim kroku mierzono podobieństwo widoku syntezowanego z różnych par widoków rzeczywistych.

Wynik eksperymentu dla każdej testowanej sekwencji przedstawiono w postaci wykresu, którego argumentami są indeksy widoków rzeczywistych użytych do syntezy. Położenie widoku syntezowanego oznaczono ciągłą czarną linią. Jakość syntezy reprezentowana jest przez kolor poszczególnych punktów wykresu – najlepsza jakość uzyskiwana dla danej sekwencji wielowidokowej reprezentowana jest kolorem zielonym, jakość najgorsza – czerwonym. Zachowano skalę liniową, a więc dla każdej sekwencji kolor punktów wykresu zmienia się proporcjonalnie do zmiany jakości syntezowanego widoku. Dokładne wartości PSNR dla widoków wirtualnych syntezowanych z różnych par widoków rzeczywistych załączono w aneksie A1.





Jak zaprezentowano na rysunku 4.7, przeprowadzony eksperyment wykazał, iż dla całego zbioru sekwencji testowych lepszą jakość widoku wirtualnego uzyska się syntezując go na podstawie najbliższego lewego i prawego widoku rzeczywistego aniżeli używając do syntezy widoków dalszych.

Przedstawione wyniki pokazują łączny wpływ na jakość syntezy trzech czynników: ograniczonej rozdzielczości obrazu, nielambertowskich odbić oraz powierzchni wspólnych obszarów sceny widocznych w widokach rzeczywistych i wirtualnym. Rzecz jasna, wspólnych obszarów sceny takich jest tym mniej, im dalej od kamery wirtualnej umieszczone są kamery rzeczywiste. Tym samym nie jest pewne, czy przedstawiony spadek jakości przy użyciu dalszych widoków rzeczywistych spowodowany jest omówionymi we wcześniejszej części rozdziału problemami, czy też wzrostem powierzchni uzupełnianych obszarów w widoku wirtualnym. W celu wyeliminowania wpływu trzeciego z czynników przeprowadzono eksperyment, w którym PSNR liczono tylko dla punktów

bezpośrednio przerzutowanych z widoków rzeczywistych (1.7). Rezultaty przedstawiono na rysunku 4.8. Dokładne wartości PSNR umieszczono w aneksie A1.



Rysunek 4.8. Jakość widoku wirtualnego syntezowanego z różnych par widoków rzeczywistych, wartość PSNR wyznaczana tylko dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych), (dla sekwencji *BBB Flowers* i *BBB Butterfly* widokiem odniesienia był widok 45, dla pozostałych sekwencji widok 4)

W większości przypadków, najlepsza jakość uzyskiwana jest w przypadku użycia sąsiednich widoków rzeczywistych. W odróżnieniu jednak od wyników przedstawionych na rysunku 4.7, w przypadku estymacji miary PSNR wyłącznie dla przerzutowanych punktów zdarzają się sytuacje, gdy wartość PSNR dla dalszej pary widoków rzeczywistych jest zbliżona, bądź nawet większa, niż dla pary widoków sąsiednich. Dla przykładu, w sekwencji *Poznan_Service2* PSNR liczony dla punktów przerzutowanych z pary widoków 5 i 9 wynosi 26,73 dB przy 26,78 dB dla pary widoków sąsiednich, a więc 3 i 5. W przypadku sekwencji *Soccer Arc* wartość PSNR dla widoków 2 i 3 wynosi 23,26 dB i jest o 0,55 dB większa, niż dla pary widoków sąsiednich (3 i 5).

Należy jednak podkreślić, iż w przypadku sekwencji *Poznan_Service2* wielkość niezsyntezowanych obszarów (a więc obszarów, dla których nie liczono PSNR) wynosiła 12% i 3% (wartości uśrednione dla 250 ramek sekwencji) powierzchni obrazu, odpowiednio dla syntezy z widoków 5 i 9 oraz 3 i 5. Dla sekwencji *Soccer Arc* obszary niezsyntezowane zajmują 6% dla syntezy z widoków sąsiednich i aż 20% (wartości uśrednione dla 250 ramek sekwencji) dla pary widoków 2 i 3. Tym samym, ze względu na zdecydowanie mniejszą powierzchnię odsłonięć, lepiej jest do syntezy widoku wirtualnego użyć bliższych widoków rzeczywistych.

Z przeprowadzonych w punkcie 4.2 rozważań oraz eksperymentu można wysnuć wniosek, iż w przypadku syntezy przy użyciu dwóch widoków rzeczywistych, najlepszą jakość widoku wirtualnego uzyska się używając najbliższego lewego i najbliższego prawego widoku rzeczywistego.

4.3. ZALETY SYNTEZY Z UŻYCIEM INFORMACJI Z DALSZYCH WIDOKÓW

Zgodnie z rozważaniami przeprowadzonymi w poprzednim punkcie, widok wirtualny ma najwyższą jakość, jeżeli do jego syntezy używany jest najbliższy lewy i najbliższy prawy widok rzeczywisty. Jednakże, widok wirtualny syntezowany z wyłącznie dwóch widoków rzeczywistych zawiera obszary niewidoczne w żadnej z sąsiednich kamer. W przypadku syntezy z dwóch widoków obszary te są uzupełniane na podstawie obszarów przerzutowanych.

Istnieje wiele technik uzupełniania obrazu, które mają różną przydatność dla syntezy widoków wirtualnych w rzadkich systemach wielokamerowych. Przegląd znanych rozwiązań w zakresie uzupełniania obrazu przedstawiono w rozdziale 2.3.

Niezależnie od efektywności techniki uzupełniania, część obszarów widoku wirtualnego jest niemożliwa do prawidłowego uzupełnienia. Przykładową sytuację, w której uzupełnianie nie da prawidłowych rezultatów pokazano na rysunku 4.9. Po lewej stronie przedstawiono widok z góry na scenę, w której znajdują się trzy kolorowe kuliste obiekty. Obiekty te są rejestrowane przez trzy kamery rzeczywiste: (i + 1), (i + 2) oraz (i + 3). W zaprezentowanym systemie syntezowany będzie widok (i + 2,5). Zarówno w lewym, jak i w prawym sąsiednim widoku rzeczywistym widoczne są tylko 2 obiekty, tymczasem w widoku wirtualnym powinien znaleźć się również fragment trzeciego, zielonego obiektu. W przypadku syntezy widoku wirtualnego na podstawie widoków sąsiednich obszar, w którym powinien znajdować się zielony obiekt nie zostanie przerzutowany z widoków rzeczywistych (czarny obszar na rysunku 4.9). W najlepszym wypadku obszar ten zostanie uzupełniony informacją z tła. Jednakże, jeśli do wypełnienia niezsyntezowanych obszarów wykorzysta się informację z widoku (i + 1), również zielony obiekt zostanie przerzutowany, a jakość widoku wirtualnego wzrośnie. Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagania



Rysunek 4.9. Idea wypełniania odsłonięć z dalszych widoków rzeczywistych; po lewej: przykładowa scena (widok z góry), po prawej: w górnym rzędzie widoki rzeczywiste, w dolnym rzędzie widok wirtualny (*i* + 2,5) syntezowany na podstawie (*i* + 2,5) oraz (*i* + 2,5) bez uzupełniania (po lewej) i z uzupełnianiem (na środku); po prawej widok (*i* + 2,5) syntezowany na podstawie wszystkich trzech widoków rzeczywistych

W tabeli 4.3 pokazano, jaki procent powierzchni widoku wirtualnego jest widoczny w dwóch, czterech i wszystkich widokach rzeczywistych.

	Obszar widoczny w widokach rzeczywistych [%]						
Sekwencja	2 sąsiednie widoki rzeczywiste	4 sąsiednie widoki rzeczywiste	Wszystkie widoki rzeczywiste				
BBB Butterfly	89,70	93,67	96,60				
BBB Flowers	86,13	92,80	95,50				
Poznan_Blocks	91,18	94,69	96,31				
Poznan_Blocks2	93,02	97,66	99,32				
Poznan_Fencing2	96,72	98,93	99,52				
Poznan_Service2	96,62	99,07	99,71				
Ballet	98,05	99,34	99,68				
Breakdancers	98,49	99,30	99,40				
Soccer Arc	93,57	96,35	97,26				
Soccer Linear	99,58	99,78	99,82				
Poznan_Carpark	99,06	99,57	99,71				
Poznan_Street	99,55	99,80	99,83				
Średnio	95,14	97,58	98,56				

Tabela 4.3. Wielkość obszaru widoku wirtualnego widocznego w co najmniej jednym widoku rzeczywistym; dla każdej sekwencji wyznaczona dla widoku wirtualnego syntezowanego w miejscu centralnego widoku rzeczywistego: widok 45 dla sekwencji *BBB Butterfly* i *BBB Flowers*, widok 4 dla pozostałych sekwencji

W lewej kolumnie tabeli 4.3 umieszczono procent punktów widoku wirtualnego widocznych w co najmniej jednym z dwóch sąsiednich widoków rzeczywistych. W środkowej kolumnie – procent punktów widoku wirtualnego widocznych w co najmniej jednym z czterech najbliższych widoków rzeczywistych. W kolumnie prawej umieszczono procent punktów widoku wirtualnego widocznych w dowolnym widoku rzeczywistym.

Jak pokazano, zwiększanie liczby widoków rzeczywistych używanych do syntezy widoku wirtualnego zmniejsza powierzchnię obszarów nieprzerzutowanych. Powierzchnia obszaru widoku wirtualnego widocznego w dwóch sąsiednich widokach rzeczywistych jest średnio o niemal 2,5 punktu procentowego mniejsza, niż w przypadku użycia czterech sąsiednich rzeczywistych (a więc w przypadku dołożenia po jednym widoku z lewej i prawej strony). Użycie wszystkich dostępnych widoków rzeczywistych zwiększa tę powierzchnię o dodatkowy punkt procentowy.

Tak znaczący wzrost powierzchni widoku wirtualnego widocznej w widokach rzeczywistych spowodowany użyciem większej liczby widoków rzeczywistych pozwala przypuszczać, iż użycie większej liczby widoków rzeczywistych pozwoli zwiększyć jakość syntezowanych widoków wirtualnych.

4.4. PODSUMOWANIE

Synteza widoku wirtualnego na podstawie dalszych widoków rzeczywistych wiąże się z trzema podstawowymi problemami: powierzchnią obszarów odsłoniętych, ograniczoną rozdzielczością obrazu i nielambertowskimi odbiciami światła. Wymienione problemy negatywnie wpływają na jakość syntezowanych widoków wirtualnych. Niemniej jednak, jak pokazano w punkcie 4.3, użycie większej liczby widoków rzeczywistych znacząco zmniejsza obszar widoku wirtualnego, który był niewidoczny w żadnym z widoków rzeczywistych.

W zgodzie z tymi obserwacjami, opracowano metodę syntezy widoków opisaną w dalszych rozdziałach rozprawy. W opracowanej w ramach rozprawy metodzie, większość powierzchni widoku wirtualnego syntezowana jest na podstawie dwóch najbliższych widoków rzeczywistych, a odsłonięcia są wypełniane przy użyciu informacji z dalszych widoków rzeczywistych. Tym samym, opracowana metoda syntezy łączy zalety syntezy z najbliższych widoków rzeczywistych (punkt 4.2) z zaletami syntezy z większej liczby widoków (punkt 4.3).

Synteza widoków wirtualnych w rzadkich systemach wielokamerowych: założenia i wymagania

5. PROPONOWANA METODA SYNTEZY WIDOKÓW WIRTUALNYCH

5.1. WPROWADZENIE

Zaproponowana przez autora metoda syntezy widoków wirtualnych (MVS, *MultiView Synthesis*) wykorzystuje reprezentację MVD, a więc jej danymi wejściowymi są widoki rzeczywiste wraz z odpowiadającymi im mapami głębi oraz parametry kamer rzeczywistych i wirtualnej. Algorytm składa się z czterech głównych etapów:

- 1. Przetwarzanie pojedynczych widoków rzeczywistych.
- 2. Łączenie sąsiednich pośrednich widoków wirtualnych.
- 3. Wypełnianie odsłonięć.
- 4. Uzupełnianie widoku wirtualnego.

Podstawową przewagę opracowanej metody nad metodami znanymi z literatury (rozdział 2) zapewnia trzeci etap, czyli wypełnianie odsłonięć.



Rysunek 5.1. Ogólny schemat proponowanej metody syntezy widoków wirtualnych. Schemat bloku przetwarzania pojedynczego widoku (PPW) przedstawiony został na rysunku 5.2

Proponowana metoda syntezy widoków wirtualnych

Ogólny schemat opracowanego algorytmu przedstawiono na rysunku 5.1. Budowę bloku odpowiadającego za przetwarzanie pojedynczego widoku (oznaczonego jako PPW) omówiono w punkcie 5.2.

W ogólności, do zsyntezowania widoku wirtualnego używana jest pewna liczba n widoków rzeczywistych. Liczba ta może wahać się od 1 do N widoków, gdzie N oznacza liczbę wszystkich dostępnych widoków rzeczywistych.

Zgodnie ze sposobem oznaczania widoków w sekwencjach testowych przyjęto, iż widoki rzeczywiste numerowane będą kolejnymi liczbami całkowitymi. Na rysunku 5.1 widoki rzeczywiste ponumerowano liczbami od (i - 3) do (i + 2). Widoki wirtualne numerowane są przy użyciu liczb wymiernych spełniających następujący warunek: $id_V \in [id_L, id_R]$ w przypadku gdy $id_L < id_R$ oraz $id_V \in [id_R, id_L]$ w przypadku gdy $id_R < id_L$, gdzie id_L jest numerem widoku wirtualnego, a id_L oraz id_R są numerami najbliższego lewego i prawego widoku rzeczywistego. Przyjęto, iż widok wirtualny oznaczany jest jako (i - 0,5) w przypadku gdy odległość euklidesowa pomiędzy wirtualną kamerą a obiema kamerami rzeczywistą *i* jest mniejsza, niż odległość pomiędzy kamerą wirtualną a kamerą (i - 1), widok wirtualny oznaczany jest liczbą z przedziału (i - 0,5, i); w przeciwnym wypadku oznaczany jest on liczbą z przedziału (i - 1, i - 0,5), np. (i - 0,9).

W przypadku użycia wyłącznie jednego widoku rzeczywistego, pokazany na rysunku 5.1 algorytm upraszcza się do dwóch etapów: pierwszego i czwartego. W przypadku użycia dwóch widoków rzeczywistych pomijany jest etap trzeci. Tym samym, gdy do syntezy widoku wirtualnego używane jest $n \leq 2$ widoków rzeczywistych, metoda upraszcza się do metody znanej z literatury i powszechnie stosowanej.

W przypadku użycia n > 2 widoków rzeczywistych, algorytm przetwarzania wygląda dokładnie tak, jak pokazano na rysunku 5.1. Najpierw n widoków rzeczywistych jest niezależnie przetwarzane w celu stworzenia n pośrednich widoków wirtualnych. Następnie widoki pośrednie stworzone na podstawie najbliższego lewego i najbliższego prawego widoku rzeczywistego są łączone. W tym momencie syntezowany widok zawiera punkty przerzutowane z sąsiednich widoków, a także obszary, które w sąsiednich widokach były przysłonięte. W trzecim etapie obszary te są uzupełniane przy użyciu informacji z dalszych widoków. Na koniec uzupełniane są pozostałe odsłonięcia w widoku wirtualnym.

5.2. PRZETWARZANIE POJEDYNCZEGO WIDOKU RZECZYWISTEGO

W pierwszym etapie zaproponowanej metody syntezy widoków wirtualnych wszystkie widoki rzeczywiste przetwarzane są niezależnie. Co więcej, przetwarzane są one w identyczny sposób, niezależnie od tego, czy są to widoki sąsiednie czy też dalsze. Sposób przetwarzania pojedynczego widoku (PPW) przedstawiono na rysunku 5.2.

Dane wejściowe stanowi widok rzeczywisty wraz z odpowiadającą mu mapą głębi, a także parametry kamery rzeczywistej i wirtualnej. Poprzez "parametry" rozumiane są zarówno parametry wewnętrzne, jak i zewnętrzne obu kamer. Na wyjściu otrzymywany jest pośredni widok wirtualny i odpowiadająca mu mapa głębi, zawierające informację przerzutowaną wyłącznie z jednego widoku.

Do przerzutowania pojedynczego widoku rzeczywistego do pozycji kamery wirtualnej zastosowano syntezę wstecz, albowiem umożliwia ona przetworzenie wirtualnej mapy głębi przed rzutowaniem barwy poszczególnych punktów widoku rzeczywistego. Wobec tego, rzutowanie przeprowadzane jest w dwóch krokach: w pierwszym kroku wyznaczana jest wirtualna mapa głębi, w kroku drugim mapa ta jest używana do rzutowania próbek barwy z widoku rzeczywistego.



Rysunek 5.2. Schemat przetwarzania informacji z pojedynczego widoku rzeczywistego wraz z odpowiadającą mu mapą glębi

W pierwszym kroku tworzona jest wirtualna mapa głębi. Dla każdego punktu rzeczywistej mapy głębi tworzony jest wektor $[x_R \ y_R \ z_R \ 1]^T$ zawierający położenie danego punktu w mapie głębi (x_R, y_R) i jego głębię z_R . Następnie wektor ten jest mnożony zgodnie z (1.3) i (1.4):

$$\begin{bmatrix} x'\\y'\\z'\\w' \end{bmatrix} = \mathbf{P}_{\mathbf{V}} \cdot \mathbf{P}_{\mathbf{R}}^{-1} \cdot \begin{bmatrix} x_{R}\\y_{R}\\z_{R}\\1 \end{bmatrix},$$
$$x_{V} = \frac{x'}{w'}, \qquad y_{V} = \frac{y'}{w'}, \qquad z_{V} = \frac{z'}{w'},$$

71 / 266

gdzie $\mathbf{P}_{\mathbf{V}}$ jest macierzą projekcji widoku wirtualnego, a $\mathbf{P}_{\mathbf{R}}$ – rzeczywistego (patrz: rozdział 1.2). Dla każdego analizowanego punktu otrzymywany jest wektor zawierający współrzędne określające pozycję odpowiadającego punktu w widoku wirtualnym wraz z jego głębią $[x_V y_V z_V 1]^{\mathrm{T}}$. Współrzędne punktu (x_V, y_V) są następnie zaokrąglane zgodnie z (1.6):

$$(\widetilde{x_V}, \widetilde{y_V}) = ([x_V], [y_V]),$$

gdzie operator [·] oznacza zaokrąglenie do najbliższej liczby całkowitej.

Podczas rzutowania punktów rozpatrywana jest kwestia przysłaniania się obiektów. Jeżeli w dany punkt wirtualnej mapy głębi przerzutowane zostanie więcej punktów niż jeden, wybierany jest ten spośród nich, którego wartość głębi z_V jest najmniejsza. Tym samym wybierany jest punkt reprezentujący najbliższy obiekt.

Przykładową przerzutowaną głębię pokazano na rysunku 5.3A. Bezpośrednio po przerzutowaniu punktów z widoku rzeczywistego, w wirtualnej głębi występuje wiele pęknięć (ang. *cracks*) – wąskich obszarów, które w ogóle nie zostały przerzutowane (rysunek 5.3B), lub których głębia znacząco odstaje od głębi otoczenia (rysunek 5.3C). Ponadto, jeżeli krawędzie obiektów w rzeczywistej mapie głębi były rozmyte (rysunek 5.4A), zostaną one przerzutowane do wirtualnej mapy głębi jako seria wąskich, nieciągłych obiektów (rysunek 5.4B).



Rysunek 5.3. Wirtualna mapa glębi (A) stworzona na podstawie jednego widoku rzeczywistego i jej fragmenty (B, C) (sekwencja *Ballet*, synteza z widoku 3 do widoku 4)


Rysunek 5.4. Obiekty z rozmytymi krawędziami w rzeczywistej mapie glębi (A) i te same obiekty przerzutowane do innego widoku (B) (fragment sekwencji *Poznan_Blocks*, synteza z widoku 3 do widoku 4)

W celu usunięcia przedstawionych blędów, wirtualna mapa głębi jest filtrowana. Zastosowany algorytm filtracji wraz z osiągniętymi rezultatami przedstawiono w rozdziale 6.1. Wartości przefiltrowanej wirtualnej mapy glębi są używane do przerzutowania informacji o barwie poszczególnych punktów tworzonego widoku wirtualnego. W tym celu dla każdego punktu wirtualnej mapy glębi tworzony jest wektor zawierający jego pozycję (x_V, y_V) i glębię z_V . Następnie zgodnie z:

$$\begin{bmatrix} x'\\ y'\\ z'\\ w' \end{bmatrix} = \mathbf{P}_{\mathbf{R}} \cdot \mathbf{P}_{\mathbf{V}}^{-1} \cdot \begin{bmatrix} x_{V}\\ y_{V}\\ z_{V}\\ 1 \end{bmatrix},$$
(5.1)

$$x_R = \frac{x'}{w'}, \qquad y_R = \frac{y'}{w'},$$
 (5.2)

$$(\widetilde{x_R}, \widetilde{y_R}) = ([x_R], [y_R])$$
(5.3)

(gdzie operator $[\cdot]$ oznacza zaokrąglenie do najbliższej liczby całkowitej) wyliczana jest pozycja w widoku rzeczywistym ($\widetilde{x_R}, \widetilde{y_R}$) punktu, którego barwa zostanie skopiowana do analizowanego punktu danego pośredniego widoku wirtualnego.

5.3. ŁĄCZENIE SĄSIEDNICH POŚREDNICH WIDOKÓW WIRTUALNYCH

5.3.1. WYBÓR NAJBLIŻSZYCH WIDOKÓW RZECZYWISTYCH

Po stworzeniu pośrednich widoków wirtualnych dla wszystkich widoków rzeczywistych, dokonywane jest łączenie pośrednich widoków wyznaczonych dla najbliższego lewego i najbliższego prawego widoku. Informacja o sąsiedztwie widoków nie jest dana, należy więc wyznaczyć, które dwa widoki zostaną połączone. Różne systemy wielokamerowe charakteryzują się jednak różnym rozmieszczeniem kamer. Wyróżnić można systemy z kamerami rozmieszczonymi liniowo (rysunek 5.5A), na łuku (rysunek 5.5B), równomiernie i parami (rysunek 5.5C). Istotne jest, by dla każdego przypadku dokonać poprawnego wyboru najbliższych widoków rzeczywistych [Dziembowski'18E].

Proponowana metoda syntezy widoków wirtualnych



Rysunek 5.5. Problem wyboru najbliższych rzeczywistych kamer dla różnych systemów wielokamerowych: A – kamerami rozmieszczone liniowo i równomiernie, B – kamery rozmieszczone równomiernie na luku, C – kamery rozmieszczone parami, D – kamery rozmieszczone liniowo, wirtualne zbliżenie się do zarejestrowanej sceny; kolorem białym oznaczono kamery rzeczywiste, kolorem szarym – kamery wirtualne

Najprostszym rozwiązaniem byłoby wyznaczenie odległości euklidesowej pomiędzy rzeczywistą i wirtualną kamerą dla wszystkich kamer rzeczywistych:

$$d(i,V) = \sqrt{\left(t_{x,i} - t_{x,V}\right)^2 + \left(t_{y,i} - t_{y,V}\right)^2 + \left(t_{z,i} - t_{z,V}\right)^2},$$
(5.4)

gdzie $t_{x,V}$, $t_{y,V}$ i $t_{z,V}$ są składowymi wektora translacji kamery wirtualnej, a $t_{x,i}$, $t_{y,i}$ i $t_{z,i}$ odpowiadają translacji kamery rzeczywistej *i*. Następnie wybrane zostałyby te dwie kamery rzeczywiste, dla których odległość d(i, V) jest najmniejsza.

Takie rozwiązanie gwarantowałoby poprawny wybór najbliższych widoków w przypadku równomiernie ustawionych kamer (rysunek 5.5A i 5.5B). Kiedy kamery rozmieszczone są nierównomiernie, dla niektórych pozycji kamery wirtualnej mogłyby zostać wybrane kamery znajdującej się po tej samej jej stronie (np. dwie kamery stojące po lewej stronie, rysunek 5.5C), co zgodnie z wnioskiem płynącym z rozdziału 4.2 przełoży się na gorszą jakość syntezowanego widoku.

Sposób umożliwiający poprawny wybór najbliższej lewej i prawej kamery dla dowolnie rozmieszczonych kamer rzeczywistych wykorzystuje wzór na nierówność trójkąta. W tym podejściu najbliższa rzeczywista kamera *i* wyszukiwana jest zgodnie z (5.4). Druga najbliższa kamera rzeczywista, oznaczona jako *j* musi spełniać warunek:

$$d(j,V) \le d(j,i), \tag{5.5}$$

a więc odległość pomiędzy kamerą wirtualną V i kamerą rzeczywistą j musi być mniejsza, niż odległość między obiema wybranymi kamerami rzeczywistymi i oraz j.

Taki sposób wyznaczania najbliższej lewej i prawej kamery rzeczywistej uniezależnia wybór od ustawienia kamer rzeczywistych. Jednakże, w przypadku gdy kamera wirtualna nie zostanie umieszczona w linii z kamerami rzeczywistymi a na przykład przed nimi, symulując zbliżenie się widza do zarejestrowanej sceny (rysunek 5.5D), rezultat pokazanego rozwiązania będzie błędny. W podanym przykładzie jako pierwsza – poprawnie – zostanie wybrana kamera 1. Drugą kamerą rzeczywistą, która powinna zostać wybrana jest kamera 2, która jest najbliższą rzeczywistą kamerą stojącą po prawej stronie kamery wirtualnej V. Jednakże, d(2,V) > d(2,1), przez co jako druga najbliższa kamera zostanie błędnie wybrana kamera 3.

Biorąc pod uwagę niedoskonałości wymienionych sposobów, w ramach prac badawczych objętych rozprawą opracowano uniwersalny sposób wyboru sąsiednich kamer rzeczywistych [Dziembowski'18E]. W zaproponowanej technice poszukiwanie najbliższych kamer przeprowadzane jest w przekształconym układzie współrzędnych (rysunek 5.6), mającym swój początek na środku przetwornika kamery wirtualnej i skierowanym tak, by oś Z' pokrywała się z osią optyczną kamery wirtualnej, a osie X' i Y' były równoległe do, odpowiednio, poziomej i pionowej osi przetwornika kamery wirtualnej.



Rysunek 5.6. Przekształcenie globalnego układu współrzędnych do układu współrzędnych kamery wirtualnej

Wektor translacji kamer rzeczywistych w przekształconym układzie współrzędnych otrzymywany jest ze wzoru:

$$\mathbf{T}_{\mathbf{i}}' = \mathbf{R}_{\mathbf{V}} \cdot (\mathbf{T}_{\mathbf{i}} - \mathbf{T}_{\mathbf{V}}), \qquad (5.6)$$

gdzie $\mathbf{R}_{\mathbf{V}}$ jest macierzą rotacji kamery wirtualnej, $\mathbf{T}_{\mathbf{i}}$ – wektorem translacji *i*-tej kamery w globalnym układzie współrzędnych, a $\mathbf{T}_{\mathbf{V}}$ wektorem translacji kamery wirtualnej w globalnym układzie współrzędnych.

Następnie, osobno dla kamer cechujących się dodatnią i ujemną składową translacji t_x , wyznaczana jest odległość euklidesowa pomiędzy kamerą a początkiem układu współrzędnych. Ta z kamer o dodatniej składowej t_x , dla której odległość od początku układu współrzędnych jest najmniejsza, jest najbliższą prawą kamerą rzeczywistą. Lewa najbliższa kamera rzeczywista jest

wybierana spośród kamer cechujących się ujemną składową translacji t_x – i jest to ta spośród nich, której odległość od środka układu współrzędnych jest najmniejsza.

5.3.2. ALGORYTM ŁĄCZENIA SĄSIEDNICH POŚREDNICH WIDOKÓW WIRTUALNYCH

Po wybraniu najbliższej lewej i prawej rzeczywistej kamery następuje proces łączenia wyznaczonych dla nich pośrednich widoków wirtualnych. Po połączeniu, każdy punkt połączonego widoku wirtualnego zawierać będzie barwę przerzutowaną z widoku lewego, z widoku prawego albo wymieszaną barwę z obu widoków. Schemat łączenia widoków przedstawiono na rysunku 5.7.



Rysunek 5.7. Schemat łączenia pośrednich widoków wirtualnych:

 c_V – barwa punktu w polączonym widoku wirtualnym; c_L , c_R – barwa punktu w lewym i prawym pośrednim widoku wirtualnym; z_L , z_R – glębia punktu przerzutowanego z lewego i prawego widoku rzeczywistego (odległość między punktem w przestrzeni i plaszczyzną przetwornika kamery wirtualnej)

Dla każdego punktu sprawdzane jest, czy został on przerzutowany z lewego lub prawego widoku rzeczywistego (1.7). Jeżeli został on przerzutowany wyłącznie z widoku lewego, do analizowanego punktu widoku wirtualnego zostanie skopiowana barwa z lewego pośredniego widoku wirtualnego. Analogicznie, jeżeli punkt został przerzutowany tylko z prawego widoku rzeczywistego,

do analizowanego punktu widoku wirtualnego skopiowana zostanie barwa z prawego pośredniego widoku wirtualnego.

Jeżeli punkt został przerzutowany z obu widoków rzeczywistych, porównywana jest wartość głębi przerzutowana z obu widoków. Jeżeli wartości głębi znacznie się różnią (a więc punkty przerzutowane z dwóch widoków reprezentują różne obiekty), do analizowanego punktu skopiowana zostanie barwa z tego widoku rzeczywistego, z którego przerzutowana została mniejsza wartość głębi – a więc barwa punktu reprezentującego bliższy obiekt. W przeciwnym wypadku, a więc jeżeli wartość głębi przerzutowanej z obu widoków rzeczywistych jest podobna, z dużym prawdopodobieństwem punkt reprezentuje ten sam obiekt w obu widokach, a więc jego barwę należy zmieszać.

W celu uniknięcia skokowej zmiany barwy obiektów przy zmianie punktu widzenia (zmianie pozycji kamery wirtualnej), zmieszana barwa punktu powinna zmieniać się w sposób proporcjonalny do odległości pomiędzy kamerą wirtualną i kamerami rzeczywistymi. Mieszanie barw zrealizowano więc poprzez średnią ważoną wyrażoną wzorem:

$$c_{V}(x,y) = \frac{c_{L}(x,y) \cdot \frac{d(R,V)}{z_{L}(x,y)} + c_{R}(x,y) \cdot \frac{d(L,V)}{z_{R}(x,y)}}{\frac{d(R,V)}{z_{L}(x,y)} + \frac{d(L,V)}{z_{R}(x,y)}},$$
(5.7)

gdzie $c_L(x, y)$ i $c_R(x, y)$ jest barwą położonego w pozycji (x, y) punktu przerzutowanego z lewego i prawego widoku rzeczywistego, $z_L(x, y)$ i $z_R(x, y)$ jego glębią (odległością punktu w przestrzeni do plaszczyzny przetwornika kamery wirtualnej), a d(L, V) i d(R, V) odległością euklidesową pomiędzy kamerą wirtualną i – odpowiednio – lewą i prawą kamerą rzeczywistą. Barwa punktu ważona jest zarówno odległością pomiędzy kamerą wirtualną i poszczególnymi kamerami rzeczywistymi (im mniejsza odległość, tym większa waga), jak i wartością glębi (im mniejsza glębia, tym większa waga).

Tak połączony widok wirtualny zawiera blędy wynikające z dwóch podstawowych przyczyn: rozmycia krawędzi obiektów w widokach rzeczywistych i wzajemnego przesunięcia krawędzi obiektów w widokach i mapach glębi. Oba zjawiska skutkują tym samym, a mianowicie rzutowaniem wąskich obszarów charakteryzujących się niepoprawną barwą (rysunek 5.8B i 5.8C).

W celu poprawy jakości syntezowanego widoku, przeprowadzana jest filtracja połączonego widoku wirtualnego. Zastosowany algorytm filtracji wraz z osiągniętymi rezultatami przedstawiono w rozdziale 6.2.

Proponowana metoda syntezy widoków wirtualnych



Rysunek 5.8. Rzutowanie blędnej barwy przy krawędziach obiektów; (B, C) – powiększone fragmenty obszaru (A) (fragment sekwencji *BBB Flowers*, synteza z widoku 32 do widoku 45)

W punkcie 5.3 opisano sposób łączenia pośrednich widoków wirtualnych. W identyczny sposób łączone są również odpowiadające im mapy głębi. Tak więc, po połączeniu sąsiednich widoków otrzymywany jest połączony widok wirtualny wraz z odpowiadającą mu mapą głębi.

5.4. Wypełnianie odsłonięć

5.4.1. WPROWADZENIE

Widok wirtualny stworzony na podstawie dwóch najbliższych widoków rzeczywistych zawiera obszary odsłonięte czyli obszary, które nie były widoczne ani w najbliższym lewym, ani w najbliższym prawym widoku rzeczywistym. W trzecim etapie zaproponowanego algorytmu syntezy obszary odsłonięte są wypełniane przy wykorzystaniu informacji z pozostałych widoków rzeczywistych.

Wykorzystanie dalszych widoków rzeczywistych do syntezy widoku wirtualnego jest znanym podejściem [Ceulemans'18, Fachada'18A, Li'18, Zhu'16B]. Opisane w literaturze metody nie pozwalają jednak na uzyskanie widoków wirtualnych dobrej jakości w przypadku systemów z rzadkim rozmieszczeniem kamer. W metodach opisanych w pracach [Ceulemans'18] i [Fachada'18A] barwa punktów w widoku wirtualnym wyznaczana jest poprzez zmieszanie barwy punktów przerzutowanych ze wszystkich widoków rzeczywistych (dalsze widoki traktowane są tak samo, jak widoki bliższe). Takie podejście jest wrażliwe na błędy w mapach głębi oraz błędy wynikające z ograniczonej rozdzielczości obrazu, a tym samym nie sprawdza się w przypadku sekwencji zarejestrowanych za pomocą rzadkich systemów wielokamerowych. Metody oparte na wypełnianiu odsłonięć za pomocą informacji z dalszych widoków rzeczywistych opisano w pracach [Li'18] i [Zhu'16B]. Metody te łączą jednak dwie

podstawowe wady: podatność na blędy w mapach glębi oraz fakt, że z dalszych widoków rzutowane są jedynie punkty tla. Podejście takie jest całkowicie blędne w przypadku skomplikowanej sceny zarejestrowanej za pomocą rzadkiego systemu wielokamerowego (np. sekwencji *BBB Flowers*). Technika wypełniania odsłonięć opracowana przez autora rozprawy nie ma wspomnianych niedoskonałości, przez co umożliwia poprawne rzutowanie punktów z dalszych widoków rzeczywistych również dla sekwencji zarejestrowanych przy użyciu rzadko rozmieszczonych kamer.

5.4.2. Opracowana technika wypełniania odsłonięć

Zgodnie ze schematem przedstawionym na rysunku 5.1, do wypełniania odsłonięć używane są pośrednie widoki wirtualne (wraz z mapami głębi) przerzutowane z dalszych widoków rzeczywistych. Podczas wypełniania odsłonięć nie jest więc konieczne wykonywanie kolejnych rzutowań.

Podczas wypełniania analizowane są wyłącznie te punkty połączonego widoku wirtualnego, które nie zostały przerzutowane z widoków sąsiednich. Dla każdego z nich sprawdzany jest podstawowy warunek – czy punkt został przerzutowany (patrz: (1.7)) z co najmniej jednego widoku rzeczywistego.

Jeżeli powyższy warunek nie jest spełniony, dany punkt nie jest wypełniany. Jeżeli punkt został przerzutowany wyłącznie z jednego widoku, jego barwa jest wprost kopiowana. W przypadku, gdy punkt został przerzutowany z większej liczby widoków, porównywana jest wartość przerzutowanej głębi dla każdego z nich. Spośród wszystkich kandydatów wybierany jest zbiór punktów oznaczany jako P. Zbiór ten zawiera te punkty, które reprezentują najbliższy z obiektów (punkty charakteryzujące się najmniejszą wartością głębi).

Dla przykładu, rozważmy przypadek sześciu pośrednich widoków wirtualnych i odpowiadających im map głębi. Wartość głębi dla poszczególnych kandydatów (odpowiadających sobie punktów w kolejnych pośrednich widokach wirtualnych: 1, 2, 3, 4, 5 i 6) wyniosła: 50, 51, x, 50, 80, 82 (gdzie "x" oznacza, iż w danym widoku pośrednim (widoku 3) analizowany punkt nie został zsyntezowany). Wartości głębi 50, 51 i 50 odpowiadają obiektowi bliższemu, niż wartości 80 i 82. Takie wartości głębi w poszczególnych widokach oznaczają, iż bliższy z obiektów nie był w ogóle widoczny w ostatnich dwóch widokach, w których w jego miejscu zsyntezowano obiekt dalszy. W przedstawionym przypadku barwa punktu w widoku wirtualnym będzie zmieszaną barwą punktów z widoku pierwszego, drugiego i czwartego.

Jeżeli zbiór P zawiera wyłącznie jeden punkt (a więc głębia w jednym widoku była zdecydowanie mniejsza, niż w pozostałych widokach), do widoku wirtualnego kopiowana jest barwa punktu z danego widoku. W innych przypadkach barwa punktu w wypełnianym obszarze odsłoniętym wyznaczana jest poprzez mieszanie barwy przerzutowanej z kilku widoków rzeczywistych. Tak jak w przypadku mieszania barw z najbliższych widoków (5.7), należy zapewnić łagodną zmianę barwy

Proponowana metoda syntezy widoków wirtualnych

obiektów podczas zmiany punktu widzenia. Tym samym, mieszanie barw z różnych widoków jest realizowane poprzez średnią ważoną wyznaczaną ze wzoru:

$$c_{V}(x,y) = \frac{\sum_{i \in P} c_{i}(x,y) \cdot \frac{D - d(i,V)}{z_{i}(x,y)}}{\sum_{i \in P} \frac{D - d(i,V)}{z_{i}(x,y)}},$$
(5.8)

gdzie $c_i(x, y)$ jest barwą analizowanego punktu dla *i*-tego widoku, $z_i(x, y)$ jego glębią, d(i, V) odległością między kamerą wirtualną a *i*-tą kamerą rzeczywistą, a *D* jest sumą odległości wszystkich kamer rzeczywistych do kamery wirtualnej:

$$D = \sum_{i \in \mathcal{P}} d(i, V) .$$
(5.9)

Powyższy sposób postępowania dobrze sprawdza się dla sekwencji z idealnymi mapami głębi (na przykład dla sekwencji syntetycznych, takich jak *BBB Butterfly* i *BBB Flowers*). W przypadku sekwencji z algorytmicznie wyznaczonymi mapami głębi, w których występują różnego rodzaju błędy i niedoskonałości, sposób postępowania musiał zostać zmodyfikowany. Dla każdego punktu sprawdzane jest, **czy punkt został przerzutowany z co najmniej k widoków rzeczywistych**, gdzie k jest pewną liczbą naturalną. Jeżeli dany punkt obszaru odsłoniętego był widoczny w mniej niż k widoków, nie jest on wypełniany z innych kamer. W przeciwnym wypadku, tak jak wcześniej opisano, tworzony jest zbiór P zawierający te punkty, dla których wartość głębi była najmniejsza. Jeżeli zbiór P zawiera k lub mniej punktów, analizowany punkt widoku wirtualnego nie jest wypełniany, jeżeli jest większy, barwa analizowanego punktu jest liczona zgodnie z (5.8).

Im większa wartość parametru k, tym więcej dalszych widoków rzeczywistych, w których dany punkt musiał mieć podobną głębię jest potrzebne, aby punkt został uzupełniony. Logicznym jest, iż w przypadku systemów z rzadkim rozmieszczeniem kamer, gdzie liczba widoków rzeczywistych jest mocno ograniczona, a scena rejestrowana była pod różnymi kątami, k nie może być zbyt duże. W przeciwnym wypadku, powierzchnia obszarów odsłoniętych wypełnionych z dalszych kamer drastycznie maleje. Eksperymentalnie stwierdzono, iż k = 2 wystarcza, by uniknąć większości błędów spowodowanych niepoprawną głębią, jednocześnie nie zmniejszając zanadto powierzchni wypełnianych obszarów.

W punkcie 5.4 opisano sposób wypełniania widoku wirtualnego. Należy podkreślić, iż wraz z widokiem wirtualnym, wypełniana jest wirtualna mapa głębi, a poszczególne próbki tejże mapy głębi wypełniane są w identyczny sposób, jak próbki widoku wirtualnego.

5.4.3. WYNIKI EKSPERYMENTALNE

Jak pokazano w tabeli 4.3, użycie większej liczby widoków zwiększa powierzchnię widoku wirtualnego, która może zostać przerzutowana z widoków rzeczywistych. W niniejszym punkcie pokazano, iż wypełnianie odsłonięć z dalszych widoków pozwala też na uzyskanie lepszej jakości widoku wirtualnego niż w przypadku uzupełniania tychże obszarów. Aby pokazać wpływ wypełniania odsłonięć z dalszych widoków na jakość widoków wirtualnych przeprowadzono eksperyment.

Dla każdej sekwencji wybrano widok odniesienia (analogicznie jak w przypadku wyników zaprezentowanych w tabeli 4.3, dla obu sekwencji *BBB* widok 45, dla pozostałych – widok 4), a następnie dokonywano syntezy widoku wirtualnego w pozycji widoku odniesienia. Synteza widoku wirtualnego przeprowadzana była w trzech konfiguracjach:

- 1. Synteza z dwóch sąsiednich widoków rzeczywistych i uzupełnianie odsłonięć na podstawie otoczenia przy użyciu oprogramowania odniesienia VSRS.
- Synteza z dwóch sąsiednich widoków rzeczywistych i uzupełnianie odsłonięć na podstawie otoczenia przy użyciu autorskiej metody MVS.
- 3. Synteza z dwóch sąsiednich widoków rzeczywistych i wypełnianie odsłonięć informacją z dalszych widoków przy użyciu autorskiej metody MVS.

Tabela 5.1. Porównanie wypełniania odsłonięć z użyciem informacji z dalszych widoków i uzupełniania; PSNR liczony **tylko dla obszarów odsłoniętych** i widocznych w dalszych widokach, średnia dla wszystkich widoków; dla każdej sekwencji pogrubiono największą wartość PSNR

	PSNR [dB]				
Sekwencja	Uzupełnianie (VSRS)	Uzupełnianie (MVS)	Wypełnianie odsłonięć (MVS)		
BBB Butterfly	29,11	27,68	35,45		
BBB Flowers	17,87	20,97	22,51		
Poznan_Blocks	24,10	24,46	23,48		
Poznan_Blocks2	19,01	18,76	27,63		
Poznan_Fencing2	27,26	27,84	29,23		
Poznan_Service2	24,80	24,89	24,97		
Ballet	27,21	27,10	28,05		
Breakdancers	24,40	24,41	24,43		
Soccer Arc	16,70	16,60	20,49		
Soccer Linear	32,56	32,52	32,66		
Poznan_Carpark	20,00	19,86	20,16		
Poznan_Street	23,22	23,33	23,70		
Średnio	23,85	24,04	26,01		

Proponowana metoda syntezy widoków wirtualnych

W tabeli 5.1 porównano jakość widoków wirtualnych syntezowanych z użyciem zaproponowanej techniki wypełniania z dalszych widoków z jakością widoków syntezowanych bez wykorzystania informacji z dalszych widoków. Przedstawione w tabeli 5.1 wyniki wyznaczone zostały wyłącznie dla obszarów odsłoniętych (nieprzerzutowanych z 2 sąsiednich widoków – patrz: (1.8)).

Należy zauważyć, że tylko dla jednej spośród 12 sekwencji testowych (*Poznan_Blocks*), wypełnianie odsłonięć z użyciem informacji z dalszych widoków nie poprawiło jakości wypełnianych obszarów, co spowodowane jest błędami w mapach głębi, skutkującymi przerzutowaniem z dalszych widoków błędnej informacji.

W poszczególnych sekwencjach różna jest powierzchnia odsłonięć (tabela 4.2), a więc różny jest także wpływ wypełniania z dalszych widoków na ogólną jakość syntezowanego obrazu. Porównanie jakości widoków syntezowanych wyłącznie z dwóch sąsiednich widoków z jakością widoków syntezowanych przy użyciu informacji z wszystkich widoków rzeczywistych przedstawiono w tabeli 5.2. W tabeli nie zamieszczono wyników dla oprogramowania VSRS ze względu na fakt, iż pozostałe części widoku są inaczej przetwarzane w opracowanej metodzie i oprogramowaniu odniesienia, a tym samym porównanie jakości wypełniania z innych widoków dla różnie syntezowanych widoków byłoby niemiarodajne.

	PSNR [dB]			
Sekwencja	Uzupełnianie	Wypełnianie		
	(MVS)	odsłonięć (MVS)		
BBB Butterfly	32,48	33,60		
BBB Flowers	24,76	25,87		
Poznan_Blocks	25,12	25,07		
Poznan_Blocks2	29,35	31,23		
Poznan_Fencing2	30,46	30,63		
Poznan_Service2	25,19	25,47		
Ballet	30,20	30,21		
Breakdancers	31,96	31,97		
Soccer Arc	22,65	22,68		
Soccer Linear	34,20	34,44		
Poznan_Carpark	33,74	33,80		
Poznan_Street	35,44	35,46		
Średnio	29,63	30,04		

Tabela 5.2. Porównanie wypełniania odsłonięć z użyciem informacji z dalszych widoków i uzupełniania; PSNR liczony **dla całego widoku wirtualnego**, średnia dla wszystkich widoków; dla każdej sekwencji pogrubiono największą wartość PSNR

Jak pokazano, użycie informacji z dalszych widoków rzeczywistych do uzupełniania odsłonięć pozwala na osiągnięcie lepszej jakości syntezy (wyłączając odznaczającą się błędnymi mapami głębi sekwencję *Poznan_Blocks*) – tym lepszej, im większa była łączna powierzchnia odsłonięć i im bardziej skomplikowany fragment sceny był przysłonięty w sąsiednich widokach.

Jak zaprezentowano w tabelach 5.1 i 5.2, niemal w każdym przypadku wypełnianie odsłonięć punktami przerzutowanymi z dalszych widoków rzeczywistych pozwala uzyskać lepszą jakość syntezowanego widoku.

5.5. UZUPEŁNIANIE WIDOKU WIRTUALNEGO

Zsyntezowany widok wirtualny, pomimo wypełnienia obszarów odsłoniętych informacją z innych kamer, wciąż zawiera obszary zawierające punkty nieprzerzutowane (1.8) – niewidoczne w żadnym z rzeczywistych widoków. Powierzchnia takich obszarów nie jest jednak duża, jak pokazano w tabeli 4.3, w zależności od sekwencji testowej waha się ona od 0,2% do 4,5%, średnio osiągając 1,5% powierzchni całego widoku wirtualnego.

Jako że powierzchnia obszarów nieprzerzutowanych nie jest duża, zastosowano uzupełnianie dyfuzyjne (rozdział 2.3.1), w którym uzupełniane obszary wypełniane są na podstawie bezpośredniego otoczenia. Co istotne, zastosowana technika bierze pod uwagę głębię próbek otoczenia, aby odsłonięte obszary uzupełnić barwą tła, a nie obiektów przysłaniających.

W pierwszym kroku dla każdego zamkniętego (otoczonego punktami przerzutowanymi) odsłoniętego obszaru wybierane są te spośród otaczających go punktów, które odznaczają się największą glębią, albowiem jedynie one będą wykorzystywane do wypełnienia analizowanego obszaru.

Dla każdego punktu uzupełnianego obszaru sprawdzany jest warunek, czy bezpośrednio sąsiaduje on z jednym z wybranych punktów. Jeżeli tak, analizowany punkt uzupełniany jest uśrednioną barwą punktów leżących w jego najbliższym otoczeniu (z pominięciem punktów charakteryzujących się mniejszą głębią). Aby uniknąć propagacji błędów, tak uzupełniony punkt nie bierze udziału w uzupełnianiu kolejnych punktów.

Po uzupelnieniu wszystkich możliwych punktów sprawdzane jest, czy w obrazie znajdują się jeszcze nieuzupelnione punkty. Jeżeli tak, rozpoczynana jest kolejna iteracja uzupelniania, w której nieuzupelnione punkty są uzupelniane na podstawie punktów uzupelnionych w poprzedniej iteracji. Sposób działania zastosowanej techniki przedstawiono na rysunku 5.9.

Proponowana metoda syntezy widoków wirtualnych



Rysunek 5.9. Iteracyjne uzupełnianie odsłonięć (oznaczonych kolorem białym) z uwzględnieniem głębi w lewej kolumnie fragment przykładowego widoku, w kolumnie prawej odpowiadająca mu mapa głębi (im ciemniejszy kolor, tym większa głębia); czerwoną ramką oznaczono punkty używane do uzupełnienia odsłonięć

W przedstawionym na rysunku 5.9 przykładzie widok wirtualny zawierał obszar odsłonięty graniczący z bliższym, niebieskim obiektem z jednej strony i z czerwono-pomarańczowym tłem ze strony drugiej. W pierwszym kroku wybrano te punkty otaczające uzupełniany obszar, które miały największą wartość głębi (na rysunku oznaczone czerwoną ramką). Następnie uzupełniono wszystkie punkty sąsiadujące z punktami zaznaczonymi na czerwono. W kolejnej iteracji do uzupełnienia używane są uprzednio uzupełnione punkty (oznaczone czerwoną ramką). Algorytm działa do momentu, gdy w obrazie nie pozostaną żadne nieuzupełnione punkty – w podanym przypadku wymaga to czterech iteracji.

Główną zaletą zastosowanego rozwiązania jest uwzględnianie informacji o głębi punktów, co skutkuje uzupełnianiem odsłonięć barwą tła (przysłoniętego w widokach rzeczywistych a odsłoniętego w syntezowanym widoku wirtualnym), a nie bliższych obiektów. Tym samym, uzupełnianie nie zwiększa sztucznie powierzchni bliskich obiektów (nie "rozlewa" ich w przylegających do nich uzupełnianych obszarach).

5.6. PODSUMOWANIE

W rozdziale 5. opisano proponowaną metodę syntezy widoków wirtualnych z wypełnianiem odsłonięć przy użyciu informacji z dalszych widoków rzeczywistych. Przedstawiono, jak z widoków rzeczywistych i odpowiadających im map głębi stworzyć widok wirtualny.

W dalszych rozdziałach rozprawy skupiono się na opracowanych technikach dodatkowo poprawiających jakość syntezowanych widoków wirtualnych:

- technice filtracji rozmytych krawędzi w widokach rzeczywistych i mapach głębi,
- sposobie rzutowania punktów do widoku wirtualnego, wymuszającego ciągłość obiektów,
- sposobie wstępnego zwiększania rozdzielczości widoków rzeczywistych i map głębi,
- technikach korekcji niespójności barwnej widoku wirtualnego.

Proponowana metoda syntezy widoków wirtualnych

6. PROPONOWANA TECHNIKA FILTRACJI PRZERZUTOWANYCH PUNKTÓW

6.1. FILTRACJA POŚREDNIEJ WIRTUALNEJ MAPY GŁĘBI

Jak napisano w rozdziale 5.2, po przerzutowaniu punktów do pośredniej wirtualnej mapy glębi, w mapie tej znajdują się liczne blędy, a więc bez przetworzenia nie umożliwi ona poprawnego rzutowania barwy punktów z rzeczywistych widoków do widoku wirtualnego. Jest to spowodowane przez dwa główne problemy opisane w dalszej części rozdziału, a mianowicie:

- Pęknięcia wąskie (o szerokości jednego okresu próbkowania) obszary zawierające punkty nieprzerzutowane (1.8) lub punkty o błędnej głębi. Ich powstawanie spowodowane jest ograniczoną rozdzielczością obrazu, co skutkuje zaokrąglaniem pozycji rzutowanych punktów do pełnych wielokrotności okresu próbkowania (rys. 1.3).
- 2. Błędy spowodowane niedoskonalościami rzeczywistych map głębi, głównie rozmyciem krawędzi w tychże mapach.

W celu rozwiązania obu problemów, zaproponowano technikę filtracji pośredniej wirtualnej mapy glębi. Schemat zaproponowanego algorytmu filtracji przedstawiono na rysunku 6.1.



Rysunek 6.1. Wybór sposobu filtracji analizowanego punktu: z_A – glębia analizowanego punktu; z_L , z_R – glębia punktów z lewej i prawej strony analizowanego punktu;

 Z_T , Z_B – glębia punktu powyżej i poniżej analizowanego punktu

Analizowane są kolejno wszystkie punkty pośredniej wirtualnej mapy glębi. W pierwszym kroku sprawdzane jest, czy analizowany punkt został przerzutowany z widoku rzeczywistego (1.7). Jeżeli nie, wykonywana jest filtracja obszarów nieprzerzutowanych. W przeciwnym wypadku glębia analizowanego punktu (z_A) porównywana jest z glębią punktów po lewej i prawej stronie (z_L i z_R), a także punktów poniżej (z_B) i powyżej (z_T). Jeżeli wartość glębi analizowanego punktu jest większa, niż glębia punktów sąsiadujących, dla analizowanego punktu wykonywana jest filtracja obszarów z blędną glębią; jeżeli jest mniejsza – eliminacja rozmytych krawędzi.

Zaznaczone fioletowym kolorem bloki z rysunku 6.1 zostały opisane w dalszej części rozdziału. Techniki filtracji obszarów nieprzerzutowanych i obszarów z błędną głębią zostały opisane w punkcie 6.1.1, a technikę redukcji błędów spowodowanych rozmytymi krawędziami opisano w punkcie 6.1.2.

6.1.1. PĘKNIĘCIA W WIRTUALNEJ MAPIE GŁĘBI

W przypadku systemów z gęstym, liniowym rozmieszczeniem kamer, pęknięcia w wirtualnej mapie głębi nie są znaczne (rysunek 6.3). Znacznie większe wyzwanie stanowią pęknięcia w przypadku rzadkich systemów wielokamerowych o nieliniowym rozmieszczeniu kamer (rysunki 6.2 i 6.4).



Rysunek 6.2. Pęknięcia w wirtualnych mapach glębi: sekwencje syntetyczne; sekwencje *BBB Flowers* (A) i *BBB Butterfly* (B), synteza widoku 45 na podstawie widoku 32

Należy zauważyć, iż pęknięcia w wirtualnej mapie głębi powstają niezależnie od tego, jaką jakość mają rzeczywiste mapy głębi, choć ich ilość w przypadku map głębi wyznaczanych w sposób algorytmiczny jest większa. Rezultat dla sekwencji syntetycznych pokazano na rysunku 6.2, dla sekwencji naturalnych – na rysunkach 6.3 i 6.4. Dla przykładu, w sekwencji *Poznan_Fencing2* na stojącym z lewej strony plakacie pojawiają się dodatkowe pęknięcia wynikające z faktu, iż głębia plakatu nie jest poprawna – grafika widoczna na plakacie "odstaje" od płaszczyzny plakatu (rysunek 6.4B).



Rysunek 6.3. Pęknięcia w wirtualnych mapach głębi: sekwencje zarejestrowane systemami z liniowym rozmieszczeniem kamer; sekwencje *Poznan_Carpark* (A) i *Poznan_Street* (B), synteza widoku 4 na podstawie widoku 3



Rysunek 6.4. Pęknięcia w wirtualnych mapach glębi: sekwencje zarejestrowane systemami z lukowym rozmieszczeniem kamer; sekwencje *Poznan_Blocks* (A), *Poznan_Fencing2* (B), *Poznan_Blocks2* (C) i *Ballet* (D), synteza widoku 4 na podstawie widoku 3

6.1.1.1. FILTRACJA OBSZARÓW NIEPRZERZUTOWANYCH

W pierwszym kroku usuwane są obszary nieprzerzutowane. Algorytm usuwania takich obszarów przedstawiono na rysunku 6.5. Dla czytelności schematu wprowadzono zmienne logiczne P_L , P_R , P_T i P_B . Przyjęto, iż P_L ma wartość logiczną "prawda", kiedy punkt po lewej stronie został przerzutowany (1.7), a "falsz", gdy nie został przerzutowany (1.8). Analogiczną regulę przyjęto

w przypadku oznaczeń P_R , P_T i P_B oznaczających odpowiednio, czy przerzutowany został punkt po prawej, powyżej i poniżej analizowanego punktu.





 z_A – glębia analizowanego punktu; z_L , z_R – glębia punktów z lewej i prawej strony analizowanego punktu; z_T , z_B – glębia punktu powyżej i poniżej analizowanego punktu; P_L , P_R , P_T i P_B przyjmują wartość logiczną "prawda", jeżeli punkt (po lewej, prawej itd.) został przerzutowany z widoku rzeczywistego i "falsz" w przeciwnym wypadku

Dla każdego nieprzerzutowanego punktu pośredniej wirtualnej mapy głębi sprawdzane są dwa warunki:

- 1. Czy przerzutowany został zarówno lewy, jak i prawy sąsiedni punkt.
- 2. Czy przerzutowany został zarówno punkt poniżej, jak i powyżej.

Jeżeli którykolwiek z podanych warunków został spełniony, analizowany punkt jest wypełniany. W przypadku spełnienia wyłącznie pierwszego warunku punkt otrzymuje uśrednioną wartość głębi lewego (z_L) i prawego punktu (z_R) (lub wartość większą, jeżeli obie wartości głębi różnią się bardziej, niż założony błąd ϵ). Gdy spełniony jest tylko warunek drugi do analizowanego punktu kopiowana jest uśredniona wartość głębi punktu powyżej (z_T) i poniżej (z_B) (lub większa z nich w przypadku dużej różnicy głębi obu punktów). Przy spełnieniu obu warunków analizowany punkt otrzymuje wartość głębi wyznaczoną jako medianę wartości głębi wszystkich sąsiadów.

Aby możliwe było usunięcie bardziej skomplikowanych pęknięć, konieczne jest dwukrotne przeprowadzenie przedstawionego powyżej wypelniania pośredniej wirtualnej mapy głębi (rysunek 6.6). Tym samym, po wykonaniu filtracji wszystkich punktów wirtualnej mapy głębi każdy punkt analizowany jest ponownie.



Rysunek 6.6. Iteracyjne wypełnianie pęknięć (oznaczonych kolorem białym) (fragment sekwencji *Breakdancers*, synteza z widoku 3 do widoku 4)

Opracowany algorytm ma dwie podstawowe zalety. Po pierwsze, wymaga niewielkiego nakładu obliczeń – dla każdego punktu obrazu analizowana jest jedynie glębia czterech sąsiadujących punktów (w dwóch iteracjach). Odróżnia to zaproponowaną technikę od technik znanych z literatury, w których dla każdego punktu analizowane są punkty wewnątrz okna o rozmiarze 5×5 [Do'11], wykorzystywana jest filtracja bilateralna [Mori'09], czy też przeprowadzana w czterech iteracjach filtracja medianowa [Senoh'17A]. Po drugie, w odróżnieniu od prostej filtracji [Jun-Te'15, Liu'12], wykorzystywana jest informacja o glębi, przez co pęknięcia wypełniane są barwą tła, a nie obiektów charakteryzujących się mniejszą glębią. Dzięki temu unika się niepoprawnego zwiększania powierzchni obiektów znajdujących się blisko kamer.

6.1.1.2. FILTRACJA OBSZARÓW Z BŁĘDNĄ GŁĘBIĄ

W drugim kroku wypełniane są te obszary, dla których pęknięcia w bliskich obiektach zostały wypełnione poprzez przerzutowanie głębi tła. Schemat zaproponowanego algorytmu przedstawiono na rysunku 6.7.



Rysunek 6.7. Schemat filtracji punktu z błędną głębią:

 z_A – glębia analizowanego punktu; z_L , z_R – glębia punktów z lewej i prawej strony analizowanego punktu; z_T , z_B – glębia punktu powyżej i poniżej analizowanego punktu

Dla każdego punktu pośredniej wirtualnej mapy głębi sprawdzane są dwa warunki:

- 1. Czy wartość głębi dla analizowanego punktu (z_A) jest większa, niż wartość głębi zarówno dla punktu po lewej (z_L), jak i punktu po prawej stronie (z_R).
- 2. Czy wartość glębi dla analizowanego punktu jest większa (z_A), niż wartość glębi zarówno dla punktu powyżej (z_T), jak i poniżej (z_B).

Jeżeli spełniony został warunek pierwszy, analizowany punkt otrzymuje nową wartość głębi otrzymaną poprzez uśrednienie wartości lewego i prawego sąsiada. W przypadku spełnienia warunku drugiego, nową wartością głębi analizowanego punktu będzie uśredniona wartość głębi punktu powyżej i poniżej. Gdy oba warunki są spełnione, liczona jest mediana głębi wszystkich czterech sąsiadujących punktów.

Tak jak filtracja obszarów nieprzerzutowanych, również filtracja obszarów z błędną głębią powtarzana jest dwukrotnie, umożliwiając usunięcie bardziej złożonych pęknięć (rysunek 6.8).



(fragment sekwencji *Poznan_Blocks*, synteza z widoku 3 do widoku 4)

Zaproponowany algorytm usuwania pęknięć w pośredniej wirtualnej mapie glębi wymaga jedynie czterokrotnej analizy każdego punktu obrazu, przez co algorytm wymaga mniejszego nakładu obliczeń, niż algorytmy filtracji stosowane w [Do'11, Senoh'17A].

6.1.1.3. PRZYKŁAD DZIAŁANIA

Wynik działania zaproponowanego algorytmu zaprezentowano na rysunkach 6.9 – 6.13. Dla każdego przykładu, po lewej stronie umieszczono wirtualną mapę głębi z pęknięciami, a po stronie prawej tę samą mapę z usuniętymi pęknięciami.



Rysunek 6.9. Wirtualna mapa głębi z pęknięciami (A) i bez pęknięć (B); sekwencja *BBB Flowers*, synteza widoku 45 na podstawie widoku 32



Rysunek 6.10. Wirtualna mapa glębi z pęknięciami (A) i bez pęknięć (B); sekwencja *BBB Butterfly*, synteza widoku 45 na podstawie widoku 32



Rysunek 6.11. Wirtualna mapa glębi z pęknięciami (A) i bez pęknięć (B); sekwencja *Poznan_Street,* synteza widoku 4 na podstawie widoku 3



Rysunek 6.12. Wirtualna mapa glębi z pęknięciami (A) i bez pęknięć (B); sekwencja *Poznan_Blocks*, synteza widoku 4 na podstawie widoku 3



Rysunek 6.13. Wirtualna mapa glębi z pęknięciami (A) i bez pęknięć (B); sekwencja *Ballet*, synteza widoku 4 na podstawie widoku 3

6.1.2. REDUKCJA BŁĘDÓW SPOWODOWANYCH ROZMYCIEM KRAWĘDZI W RZECZYWISTYCH MAPACH GŁĘBI

Poza opisanym w punkcie 6.1.1 usuwaniem pęknięć, w pośredniej wirtualnej mapie głębi usuwane są również te obszary, których powstanie mogło wynikać z rozmycia krawędzi w rzeczywistych mapach głębi. Obszary takie charakteryzują się niewielką szerokością (do kilku okresów próbkowania) i mniejszą od otoczenia głębią. Wirtualna mapa głębi z tego rodzaju błędami została pokazana na rysunku 6.14. Opisywane błędy są widoczne jako male jasne obszary na tle ciemniejszego, mającego większą głębię otoczenia.



Rysunek 6.14. Występowanie nieistniejących obiektów w wirtualnej mapie glębi spowodowane rozmyciem krawędzi w rzeczywistej mapie glębi (sekwencja *Poznan_Blocks* (A) i jej fragment (B), synteza z widoku 7 do widoku 4)

Eliminacja takich obszarów jest kluczową kwestią dla osiągnięcia zadowalającej jakości subiektywnej podczas wirtualnej nawigacji, co jest to spowodowane małą wartością glębi tych obszarów. Niewyeliminowanie blędów spowodowanych rozmytymi krawędziami skutkuje tym, iż niepoprawnie zsyntezowane obszary znajdują się blisko widza (mają mniejszą glębię niż otoczenie), a więc podczas wirtualnego poruszania się w scenie wydają się szybko przemieszczać, skupiając na sobie uwagę widza.

Do usunięcia opisywanych błędów zaproponowano algorytm (rysunek 6.15), w którym glębia każdego punktu wirtualnej mapy głębi porównywana jest z głębią jego sąsiadów. Opracowany algorytm pozwala na dokładniejszą eliminację błędnych obszarów, niż często stosowana filtracja medianowa [Jun-Te'15, Liu'12, Senoh'17A].



Rysunek 6.15. Schemat eliminacji rozmytych krawędzi rzeczywistych map głębi: z_A – glębia analizowanego punktu; z_L , z_R – glębia punktów z lewej i prawej strony analizowanego punktu; z_T , z_B – glębia punktu powyżej i poniżej analizowanego punktu

W celu eliminacji jak największej liczby punktów z blędną głębią, głębia każdego punktu jest porównywana z głębią punktów sąsiadujących zarówno w pionie, jak i w poziomie. Jeżeli oba sąsiadujące punkty charakteryzują się większą głębią, niż punkt analizowany, wartość głębi dla tego punktu jest usuwana, a on sam jest traktowany jako punkt nieprzerzutowany.

Wynik działania zaproponowanej techniki zaprezentowano na rysunku 6.16, pokazując te same fragmenty mapy głębi, jak na rysunku 6.14.



Rysunek 6.16. Eliminacja nieistniejących obiektów w wirtualnej mapie glębi (sekwencja *Poznan_Blocks* (A) i jej fragment (B), synteza z widoku 7 do widoku 4)

Zaprezentowane podejście ma jedną wadę: podczas eliminacji obiektów z niepoprawną głębią usunięte mogą zostać również istniejące obiekty charakteryzujące się niewielkim obszarem i mniejszą głębią niż głębia punktów w jego otoczeniu. Jednakże, eliminacja szybko przemieszczających się w scenie nieistniejących w rzeczywistości obiektów rekompensuje ewentualny brak drobnych szczegółów syntezowanego widoku [Mieloch'16].

6.2. ELIMINACJA EFEKTÓW ROZMYTYCH KRAWĘDZI W WIDOKACH RZECZYWISTYCH

W rozdziale 5.3 zwrócono uwagę na fakt, iż w pewnych przypadkach w widoku wirtualnym pojawiają się nieistniejące w rzeczywistej scenie krawędzie. Są dwie główne przyczyny ich powstawania: rozmycie krawędzi obiektów w widokach rzeczywistych (rysunek 6.18), a także przesunięcie krawędzi pomiędzy widokiem rzeczywistym a odpowiadającą mu mapą głębi (rysunek 6.17).





Rysunek 6.17. Przesunięcie krawędzi obiektu między widokiem rzeczywistym i odpowiadającą mu mapą głębi: (A): sekwencja *BBB Butterfly*, fragment widoku rzeczywistego (po lewej) i odpowiadającej mu mapy głębi (po prawej), (B): sekwencja *BBB Flowers*, fragment widoku rzeczywistego (do góry) i odpowiadającej mu mapy głębi (na dole)



Rysunek 6.18. Rozmycie krawędzi obiektu w widoku rzeczywistym: (A): sekwencja *Poznan_Fencing2*, fragment widoku rzeczywistego (do góry) i odpowiadającej mu mapy głębi (na dole), (B): sekwencja *Ballet*, fragment widoku rzeczywistego (do góry) i odpowiadającej mu mapy głębi (na dole)

Skutek pokazanych na rysunkach 6.17 i 6.18 niedoskonałości zaprezentowano na rysunku 6.19. Opisywany efekt jest najbardziej widoczny, kiedy krawędź bliższego obiektu rzutowana jest na jednolity obszar, na przykład tło. Niemniej jednak, nieistniejące krawędzie widoczne są w widoku wirtualnym również w przypadku bardziej złożonych obszarów (na przykład obszarów zawierających krawędzie).



Rysunek 6.19. Synteza dla rozmytych krawędzi w widokach rzeczywistych; fragmenty sekwencji *BBB Flowers* (A), synteza z widoku 32 do widoku 45 oraz *Ballet* (B), synteza z widoku 3 do widoku 4

W celu usuwania opisywanych blędów zaproponowano technikę, w której punkty znajdujące się w okolicach krawędzi obiektów (a więc punkty, w których otoczeniu występuje skokowa zmiana głębi) są przetwarzane w inny sposób, niż pozostałe punkty.

Jak przedstawiono w rozdziale 2.4, podejście takie nie jest rozwiązaniem nowym, jako że problem usuwania nieistniejących krawędzi w wirtualnym widoku opisywany był w licznych pracach. W odróżnieniu od technik opisywanych w literaturze, w zaproponowanym podejściu [Dziembowski'18B] dodatkowo sprawdzane są gradienty w sąsiedztwie analizowanego punktu w widoku rzeczywistym, umożliwiając rzutowanie punktów znajdujących się przy krawędziach obiektów w sytuacji, gdy ich barwa jest poprawna.

6.2.1. Algorytm filtracji

Dla każdego punktu widoku rzeczywistego estymowane są 3 gradienty, osobno dla każdej składowej Y, Cb i Cr. Gradient szacowany jest poprzez różnicę danej składowej dla analizowanego punktu i punktu następnego (po prawej stronie lub poniżej). Następnie dla każdego punktu tła graniczącego z obszarem o mniejszej glębi sprawdzane jest, czy norma jego gradientu jest większa, niż średnia norm gradientów trzech punktów sąsiadujących. Schemat zaproponowanego algorytmu przedstawiono na rysunku 6.20.



Rysunek 6.20. Schemat eliminacji rozmytych krawędzi w widokach rzeczywistych

Aby punkt tla graniczący lewostronnie z bliższym obiektem (mający po lewej stronie punkt o mniejszej wartości glębi) był rzutowany do widoku wirtualnego, musi spełniać warunek:

$$|c(x+1,y) - c(x+2,y)| < \frac{1}{3} \sum_{i=2}^{4} |c(x+i,y) - c(x+i+1,y)|, \qquad (6.1)$$

gdzie c(x, y) oznacza wartość analizowanej składowej barwnej w punkcie (x, y). W przypadku punktu tla graniczącego z bliższym obiektem prawostronnie (mającego po prawej stronie punkt o mniejszej wartości glębi), punkt nie będzie rzutowany w przypadku braku spełnienia warunku:

$$|c(x,y) - c(x-1,y)| < \frac{1}{3} \sum_{i=1}^{3} |c(x-i,y) - c(x-i-1,y)|.$$
(6.2)

W przypadku pionowej składowej gradientu oba warunki skonstruowane są analogicznie, przy czym analizowana jest pionowa, a nie pozioma składowa położenia punktów.

6.2.2. PRZYKŁAD DZIAŁANIA

Na rysunku 6.21 przedstawiono sposób wykrywania i usuwania rozmytych bądź przesuniętych krawędzi obiektów w widoku rzeczywistym. Po lewej stronie przedstawiono fragment rzeczywistej mapy glębi, pośrodku – odpowiadający jej fragment widoku rzeczywistego (jego jednej składowej, na przykład luminancji). Krawędź pomiędzy bliższym i dalszym obiektem jest ostra w mapie glębi i rozmyta w widoku rzeczywistym. Punkty tła graniczące z bliższym obiektem zaznaczono czerwoną ramką.

rzeczywista mapa głębi

widok rzeczywisty

50	50	49	48	47	30	3	2	1
51	50	49	47	46	6	4	1	2
50	49	45	43	23	3	0	2	0
48	45	41	42	20	6	0	1	1
45	48	44	45	6	4	1	3	1
47	48	47	15	3	2	1	2	1
50	48	44	5	3	0	1	1	2
48	50	30	3	0	0	2	0	3

moduł poziomej składowej gradientu

0	1	1	1	17	27	1	1	1
1	1	2	1	40	2	3	1	2
1	4	2	20	20	3	2	2	0
3	4	1	22	14	6	1	0	1
3	4	1	39	2	3	2	2	1
1	1	32	12	1	1	1	1	1
2	4	39	2	3	1	0	1	2
2	20	27	3	0	2	2	3	3

Rysunek 6.21. Eliminacja punktów na podstawie gradientu otoczenia

W przykładzie pokazano analizę obrazu wyłącznie dla poziomej składowej gradientu jednej składowej, jednakże w przypadku pozostałych gradientów sposób analizy obrazu jest analogiczny. W przypadku poziomej składowej gradientu dla każdego analizowanego punktu sprawdzana jest

wartość bezwzględna poziomej składowej gradientu jego samego i gradientu sąsiadujących (w poziomie) punktów tła – a więc, gdy bliższy obiekt znajduje się po lewej stronie (jak w przykładzie) sprawdzana jest wartość bezwzględna poziomej składowej gradientu trzech punktów znajdujących się po prawej stronie. Przykładowo, dla analizowanego punktu z ostatniego wiersza (punkt o luminancji 30) sprawdzana jest wartość bezwzględna poziomej składowej gradientu punktów o luminancji 3, 0 i 0, a więc kolejno 3, 0 i 2. W ostatnim kroku wartość bezwzględna trzech sprawdzanych sąsiednich punktów jest uśredniana (dla omawianych punktów jest to 1,67) i porównywana z wartością bezwzględną gradientu analizowanego punktu (wynoszącym 27).

Jeżeli wartość bezwzględna poziomej składowej gradientu analizowanego punktu jest większa, niż średnia dla trzech sąsiednich punktów, punkt nie będzie rzutowany do widoku wirtualnego (wartości zaznaczone na rysunku 6.21 czerwoną czcionką).

Jak napisano w punkcie 6.2.1, w obrazie analizowana jest składowa pionowa i pozioma 3 gradientów. Jeżeli analizowany punkt zostanie odrzucony na podstawie któregokolwiek z 6 analizowanych składników, nie jest on rzutowany do widoku wirtualnego.

Dzięki zastosowanemu podejściu możliwa jest eliminacja nieistniejących krawędzi w widoku wirtualnym poprzez brak rzutowania punktów w pobliżu rozmytych tudzież przesuniętych krawędzi obiektów przy jednoczesnym rzutowaniu wszystkich punktów tła posiadających poprawną barwę.

Na rysunkach 6.22 – 6.24 pokazano wpływ zaproponowanej techniki eliminacji efektów rozmytych krawędzi w widokach rzeczywistych na jakość widoków wirtualnych. Z lewej strony rysunków 6.22 – 6.24 umieszczono fragmenty widoków wirtualnych syntezowanych bez użycia zaproponowanej techniki, z prawej strony z użyciem zaproponowanej techniki.



Rysunek 6.22. Synteza dla rozmytych krawędzi w widokach rzeczywistych przed (A) i po ich eliminacji (B); fragment sekwencji *BBB Flowers*, synteza z widoku 32 do widoku 45



Rysunek 6.23. Synteza dla rozmytych krawędzi w widokach rzeczywistych przed (A) i po ich eliminacji (B); fragment sekwencji *Ballet*, synteza z widoku 3 do widoku 4





Rysunek 6.24. Synteza dla rozmytych krawędzi w widokach rzeczywistych przed (A) i po ich eliminacji (B); fragment sekwencji *Breakdancers*, synteza z widoku 3 do widoku 4

6.3. WPŁYW FILTRACJI NA JAKOŚĆ SYNTEZY

6.3.1. OBIEKTYWNY POMIAR JAKOŚCI

W celu zbadania wpływu zaproponowanej techniki filtracji na jakość syntezowanych widoków wirtualnych porównano mierzoną obiektywnie jakość widoków syntezowanych przy użyciu zaproponowanego algorytmu syntezy z włączoną i całkowicie wyłączoną filtracją. Ponadto, jakość zmierzono również dla konfiguracji, w której włączone było wypełnianie pęknięć (opisane w punkcie 6.1.1) w widoku wirtualnym, a wyłączone obie techniki filtracji krawędzi (punkty 6.1.2 i 6.2).

Rezultaty przedstawiono w tabelach 6.1 i 6.2. Dla każdej sekwencji testowej w tabeli umieszczono średnią jakość syntezowanego widoku wirtualnego. Wartość ta jest uśredniona w czasie (dla wszystkich ramek sekwencji testowej) i dla wszystkich nieskrajnych widoków rzeczywistych. Obszerniejsze wyniki, z wyszczególnieniem jakości wszystkich syntezowanych widoków, umieszczono w aneksie A2.

Tabela 6.1. Jakość syntezowanych widoków dla trzech konfiguracji MVS, średnia dla widoków wirtualnych syntezowanych w pozycji wszystkich nieskrajnych widoków rzeczywistych; pełna filtracja = wypełnianie pęknięć + eliminacja efektów rozmytych krawędzi w rzeczywistych mapach głębi

	PSNR [dB]				
Sekwencja	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
BBB Butterfly	30,63	32,93	33,58		
BBB Flowers	24,36	25,84	25,97		
Poznan_Blocks	24,68	25,20	25,21		
Poznan_Blocks2	31,20	31,52	31,62		
Poznan_Fencing2	30,50	30,64	30,69		
Poznan_Service2	25,22	25,26	25,48		
Ballet	29,82	29,95	30,09		
Breakdancers	31,23	31,48	31,63		
Soccer Arc	20,96	21,15	21,21		
Soccer Linear	34,32	34,35	34,28		
Poznan_Carpark	33,88	34,04	33,88		
Poznan_Street	35,83	35,79	35,72		
Średnio	29,39	29,85	29,95		

Dla każdej sekwencji testowej pogrubiono najlepszą otrzymaną jakość. Jak pokazano, największy wpływ na spadek jakości ma wypełnianie pęknięć. Spowodowane jest to dużą powierzchnią, jaką blędy tego typu zajmują (rysunki 6.2 - 6.4). W ogólności, powierzchnia blędów spowodowanych rozmytymi krawędziami nie jest duża, implikując niewielką poprawę mierzonej w sposób obiektywny jakości całego obrazu.

Należy zauważyć również, iż w przypadku sekwencji zarejestrowanych za pomocą gęstych systemów wielokamerowych z liniowym rozmieszczeniem kamer, dodatkowa filtracja zmniejsza jakość obiektywną syntezowanego widoku. Przyczyną jest fakt, iż zaproponowany algorytm usuwa punkty o glębi mniejszej lub większej od glębi punktów sąsiednich nie analizując, czy dany punkt został przerzutowany poprawnie, czy też nie. Tym samym, usuwane są zarówno punkty przerzutowane

poprawnie, jak i te, które zostały przerzutowane błędnie. W przypadku sekwencji zarejestrowanych przy użyciu liniowych systemów wielokamerowych, punktów przerzutowanych niepoprawnie jest mniej, co skutkuje niewielkim spadkiem PSNR.

Tabela 6.2. Jakość syntezowanych widoków dla trzech konfiguracji MVS, średnia dla widoków wirtualnych
syntezowanych w pozycji wszystkich nieskrajnych widoków rzeczywistych;
pełna filtracja = wypełnianie pęknięć + eliminacja efektów rozmytych krawędzi w rzeczywistych mapach glębi

	SSIM					
Sekwencja	Brak filtracji	Wypełnianie pęknięć	ie Pełna filtracja			
BBB Butterfly	0,934	0,959	0,965			
BBB Flowers	0,818	0,872	0,878			
Poznan_Blocks	0,749	0,786	0,797			
Poznan_Blocks2	0,851	0,859	0,861			
Poznan_Fencing2	0,874	0,880	0,880			
Poznan_Service2	0,802	0,813	0,816			
Ballet	0,840	0,846	0,853			
Breakdancers	0,838	0,845	0,846			
Soccer Arc	0,740	0,755	0,755			
Soccer Linear	0,905	0,905	0,906			
Poznan_Carpark	0,932	0,933	0,932			
Poznan_Street	0,938	0,938	0,937			
Średnio	0,852	0,866	0,869			

Jak pokazano w tabelach 6.1 i 6.2, pełna filtracja widoku wirtualnego nie zmienia jakości mierzonej metodami obiektywnymi w sposób znaczący. Skutkiem zastosowania filtracji jest jednak poprawa jakości mierzonej w sposób subiektywny – albowiem dzięki zastosowanej filtracji krawędzi w rzeczywistym widoku i odpowiadającej mu mapie głębi, w widoku wirtualnym pojawia się mniej nieistniejących w rzeczywistości, a zwracających uwagę widza obiektów.

6.3.2. SUBIEKTYWNY POMIAR JAKOŚCI

Aby zbadać wpływ zaproponowanych technik filtracji na subiektywne postrzeganie jakości syntezowanego widoku przeprowadzono również testy subiektywne, podczas których uczestnicy porównywali jakość widoków syntezowanych w dwóch konfiguracjach: z włączonym wypełnianiem pęknięć oraz włączonymi wszystkimi zaproponowanymi technikami filtracji. Ich wynik przedstawiono na rysunku 6.25.

Jak zaprezentowano, dla niemal wszystkich sekwencji testowych poprawa jakości mierzonej w sposób subiektywny jest statystycznie istotna. Wyjątkiem jest sekwencja *Soccer Arc*, dla której zaproponowana technika filtracji również powoduje poprawę jakości, jednakże na tyle małą, iż na podstawie przeprowadzonych badań jakości subiektywnej nie można stwierdzić, że poprawa ta jest statystycznie istotna.



Rysunek 6.25. Poprawa mierzonej w sposób subiektywny jakości syntezowanych widoków z dodatkową filtracją i bez; dla każdej sekwencji oznaczono 95% przedział ufności; wartości dodatnie oznaczają, że jakość widoków syntezowanych przy użyciu MVS jest lepsza, niż w przypadku VSRS

Największą poprawę otrzymano dla sekwencji *Poznan_Blocks*, która to charakteryzuje się rozmytymi krawędziami w rzeczywistych mapach głębi. Tym samym, bez dodatkowej filtracji w widoku wirtualnym pojawiają się liczne błędy degradujące jakość subiektywną.

6.4. PODSUMOWANIE

Wśród opisywanych w rozdziale 6. algorytmów wyróżnić można dwie rodzaje filtracji:

- 1. Eliminację pęknięć w wirtualnej mapie głębi.
- 2. Eliminację błędów spowodowanych rozmyciem krawędzi w widokach rzeczywistych i odpowiadających im mapach głębi.

Poprawa jakości spowodowana eliminacją pęknięć jest niepodważalna. Zaproponowany algorytm umożliwia zwiększenie wartości PSNR o średnio 0,5 dB (dla SSIM: 0,014) w porównaniu do syntezy bez eliminacji pęknięć (tabele 6.1 i 6.2).

Drugi rodzaj zaproponowanej w rozprawie filtracji, a więc eliminacja błędów wynikających z rozmycia krawędzi obiektów w rzeczywistych widokach i mapach głębi, zwiększa wartość PSNR o dodatkowe 0,1 dB (dla SSIM: 0,003). Poprawa jakości mierzonej obiektywnie nie jest duża, jednakże głównym celem stosowania zaproponowanej techniki filtracji jest eliminacja błędów zwracających uwagę widza, a więc poprawa jakości mierzonej subiektywnie.

W przypadku subiektywnego pomiaru jakości, eliminacja błędów spowodowanych rozmyciem krawędzi wyraźnie zwiększa jakość syntezowanych widoków wirtualnych. Statystycznie istotną poprawę jakości (w porównaniu do jakości widoków syntezowanych wyłącznie z eliminacją pęknięć; bez eliminacji skutków rozmycia) osiągnięto dla 11 z 12 sekwencji z badanego zbioru sekwencji testowych (rysunek 6.25).

7. PROPONOWANA TECHNIKA RZUTOWANIA PUNKTÓW ZAPEWNIAJĄCA CIĄGŁOŚĆ OBIEKTÓW

7.1. WPROWADZENIE

Opisywane w rozdziale rozwiązanie stanowi modyfikację opisanego w rozdziale 5.2 etapu rzutowania mapy glębi (fioletowy blok na rysunku 7.1).



Rysunek 7.1. Schemat przetwarzania informacji z pojedynczego widoku rzeczywistego wraz z odpowiadającą mu mapą glębi

W typowym podejściu podczas tworzenia wirtualnej mapy głębi wszystkie punkty z rzeczywistych map głębi są rzutowane do wirtualnej mapy głębi w sposób niezależny. Tym samym, jeżeli dwa sąsiadujące w rzeczywistej mapie głębi punkty po przerzutowaniu do wirtualnej mapy głębi znajdą się w większej odległości, pomiędzy nimi powstanie nieciągłość.



Rysunek 7.2. Powstawanie nieciągłości pomiędzy sąsiednimi punktami rzeczywistej mapy glębi podczas rzutowania: (A) – dla 2 obiektów, (B) – dla 1 obiektu (w obu przypadkach po lewej rzeczywista mapa glębi, po prawej wirtualna)

Proponowana technika rzutowania punktów zapewniająca ciągłość obiektów

W przypadku, gdy punkty te odpowiadały różnym obiektom sceny (jeden z punktów reprezentował bliższy obiekt, drugi – obiekt dalszy), nieciągłość ta powinna powstać (rysunek 7.2A). Jeżeli jednak oba punkty reprezentowały jeden obiekt, a większa odległość pomiędzy nimi w wirtualnej mapie głębi wynika z próbkowania obrazu, nieciągłość ta jest niepożądana (rysunek 7.2B) i powinna zostać wyeliminowana.

7.2. Opis zaproponowanej techniki

Zaproponowano, by typowe podejście do rzutowania punktów zmienić, nie rzutując wszystkich punktów niezależnie, a w grupach po trzy sąsiednie punkty. Oba podejścia zestawiono na rysunku 7.3.



Rysunek 7.3. Rzutowanie punktów z rzeczywistej do wirtualnej mapy glębi: niezależne (A) i w trójkątach (B)

7.2.1. ALGORYTM RZUTOWANIA ZAPEWNIAJĄCEGO CIĄGŁOŚĆ OBIEKTÓW

Zaproponowany algorytm składa się z pięciu głównych kroków.

W pierwszym kroku trzy sąsiadujące z sobą (tworzące trójkąt) punkty są rzutowane z rzeczywistej mapy glębi zgodnie z (1.3). W wyniku rzutowania otrzymywane są pozycje trzech punktów w pośrednim widoku wirtualnym: $A = (x_A, y_A), B = (x_B, y_B)$ i $C = (x_C, y_C)$ wraz z odpowiadającymi im wartościami glębi: z_A, z_B i z_C .

W drugim kroku sprawdzana jest glębia rzutowanych punktów. Jeżeli wartości glębi poszczególnych punktów w widoku rzeczywistym znacząco się różniły (punkty reprezentowały różne obiekty sceny), pomiędzy tymi punktami nie jest zachowywana ciągłość. Tym samym, punkty te są przetwarzane w typowy sposób, opisany w rozdziale 5.2. W przeciwnym wypadku, w celu zachowania ciągłości pomiędzy punktami, wykonywany jest krok trzeci.

W trzecim kroku zaproponowanego algorytmu wyznaczany jest prostokątny obszar ograniczający w mapie glębi odpowiadającej pośredniemu widokowi wirtualnemu:
$$\begin{aligned} x_{min} &= \min(x_A, x_B, x_C), & x_{max} &= \max(x_A, x_B, x_C), \\ y_{min} &= \min(y_A, y_B, y_C), & y_{max} &= \max(y_A, y_B, y_C). \end{aligned}$$
(7.1)

W czwartym kroku dla wszystkich punktów $V = (x_V, y_V)$ spełniających warunki:

$$x_V \in \mathbb{Z} \land x_V \in [x_{min}, x_{max}] \land y_V \in \mathbb{Z} \land y_V \in [y_{min}, y_{max}]$$
(7.2)

sprawdza się, czy punkt V znajduje się wewnątrz trójkąta o wierzchołkach (x_A, y_A) , (x_B, y_B) i (x_C, y_C) . Sprawdzenie to dokonywane jest poprzez porównanie trzech iloczynów wektorowych:

$$\vec{a} = \overrightarrow{AV} \times \overrightarrow{AB}$$
, $\vec{b} = \overrightarrow{BV} \times \overrightarrow{BC}$, $\vec{c} = \overrightarrow{CV} \times \overrightarrow{CA}$. (7.3)

Na rysunku 7.4 powyższe iloczyny zostały przedstawione w sposób graficzny. Liniami ciągłymi oznaczono wektory \overrightarrow{AB} , \overrightarrow{BC} i \overrightarrow{CA} , liniami przerywanymi: \overrightarrow{AV} , \overrightarrow{BV} i \overrightarrow{CV} , a kropkowanymi: \vec{a} , \vec{b} i \vec{c} . Wektory \vec{a} , \vec{b} i \vec{c} są prostopadle do plaszczyzny analizowanego obrazu, a ich zwrot zależy od tego, czy punkt V znajduje się wewnątrz trójkąta ABC.

Jeżeli wektory \vec{a} , \vec{b} i \vec{c} mają ten sam zwrot, punkt V znajduje się wewnątrz analizowanego trójkąta, a więc wartość glębi tego zostanie wyznaczona na podstawie wartości glębi w wierzchołkach trójkąta.



Rysunek 7.4. Sposób sprawdzania, czy punkt V znajduje się wewnątrz trójkąta ABC;
(A) jeżeli punkt V jest wewnątrz trójkąta ABC, wektory a, b i c mają ten sam zwrot,
(B) jeżeli punkt V jest poza trójkątem ABC, jeden z wektorów a, b lub c ma przeciwny zwrot do dwóch pozostałych

W piątym, ostatnim kroku, wyznaczana jest wartość glębi dla wszystkich punktów V znajdujących się wewnątrz przerzutowanego trójkąta. Glębia punktów znajdujących się wewnątrz

trójkąta powinna zmieniać się w sposób płynny. Z tego powodu, wartość glębi punktu V estymowana jest jako średnia ważona glębi trzech wierzchołków trójkąta.

7.2.2. PRZYKŁAD DZIAŁANIA

W przykładzie zaprezentowanym na rysunku 7.3 do wirtualnej mapy glębi rzutowane jest 6 punktów widoku rzeczywistego, oznaczonych literami A – F. Wartość glębi w tych punktach i parametry kamer spowodowały, iż punkty, które w rzeczywistej mapie glębi były zgrupowane i tworzyły spójny obszar, w mapie wirtualnej zostały rozmieszczone rzadziej. W przypadku typowego, niezależnego rzutowania poszczególnych punktów (rysunek 7.3A), pomiędzy punktami A – F nie została przerzutowana żadna informacja, a więc w widoku wirtualnym powstała nieciągłość.

W zaproponowanym podejściu punkty grupowane są w trójki zawierające sąsiednie punkty rzeczywistej mapy głębi, kolejno ABD, BDE, BCE i CEF, oznaczone na rysunku kolorowymi trójkątami. Po przerzutowaniu wierzchołków każdego z trójkątów sprawdzane jest, które z punktów wirtualnej mapy głębi znajdują się wewnątrz danego trójkąta (4 punkty zaznaczone na niebiesko w przypadku trójkąta ABD, 2 zielone punkty dla trójkąta BDE i 1 punkt oznaczony kolorem niebieskim dla trójkąta BCE). Następnie wartość głębi dla każdego punktu leżącego wewnątrz trójkąta estymowana jest na podstawie wartości głębi w trzech wierzchołkach trójkąta i odległości między wierzchołkami trójkąta i analizowanego punktu.

W przypadku, gdy trzy tworzące trójkąt punkty charakteryzują się znacząco różną głębią (a więc reprezentują różne obiekty sceny), wymuszenie ciągłości w widoku wirtualnym byłoby błędne. W takim przypadku w proponowanej technice punkty rzutowane są w sposób typowy, niezależny.

7.3. WYNIKI EKSPERYMENTALNE

Na rysunkach 7.5 i 7.6 porównano widoki wirtualne zsyntezowane w sposób typowy (z niezależnym rzutowaniem punktów) z widokami zsyntezowanymi przy zastosowaniu zaproponowanej techniki rzutowania punktów zapewniającej ciągłość obiektów.

Jak pokazano na rysunkach 7.5 i 7.6, rzutowanie trójkątami, poprzez zapewnienie ciągłości obiektów w syntezowanym widoku, umożliwia poprawne przerzutowanie obiektów nawet z silnie oddalonych widoków rzeczywistych (w przypadku sekwencji *BBB Butterfly* i *BBB Flowers* kąt między osiami optycznymi kamery 06 i 84 wynosi 39 stopni, kąt między osiami optycznymi skrajnych kamer dla sekwencji *Ballet* i *Breakdancers* wynosi 30 stopni.



Rysunek 7.5. Porównanie niezależnego rzutowania punktów (A, B) i rzutowania zapewniającego ciągłość obiektów (C, D); sekwencje *BBB Butterfly* (A, C) i *BBB Flowers* (B, D), synteza z widoku 06 do 84

С

D



Rysunek 7.6. Porównanie niezależnego rzutowania punktów (A, B) i rzutowania zapewniającego ciągłość obiektów (C, D); sekwencje *Ballet* (A, C) i *Breakdancers* (B, D), synteza z widoku 0 do widoku 7

Proponowana technika rzutowania punktów zapewniająca ciągłość obiektów



Rysunek 7.7. Wirtualne wejście w scenę, sekwencja *BBB Flowers* – synteza widoku z kamery skierowanej jak kamera 45, ale umieszczonej znacznie bliżej sceny (za pierwszoplanowymi kwiatami); porównanie syntezy przy niezależnym rzutowaniu punktów (A, B) i rzutowaniu zapewniającym ciągłość obiektów (C, D)

W przypadku niepoprawnych wartości mapy głębi może się zdarzyć, iż punkty, pomiędzy którymi w widoku wirtualnym powinna zostać zachowana ciągłość, pozostaną niepołączone. Taki efekt pokazany został na rysunku 7.6C i 7.6D, gdzie dla pewnych obszarów ściany czy podłogi w widoku wirtualnym pojawiają się nieciągłości. Takiego problemu nie ma dla sekwencji z poprawnie wyznaczonymi mapami głębi (rysunek 7.5C i 7.5D).

Rzutowanie zapewniające ciągłość obiektów doskonale sprawdza się w jeszcze jednym przypadku – podczas wirtualnego "podejścia" do sceny, a więc w momencie, gdy użytkownik zbliża się do zarejestrowanych obiektów. Taki scenariusz przedstawiono na rysunkach 7.7 i 7.8.

W przypadku widoku syntezowanego dla sekwencji *BBB Flowers* zapewnienie ciągłości obiektów pozwoliło na zachowanie kształtu widocznych kwiatów czy też poprawne odwzorowanie króliczego futra. Dla sekwencji *Poznan_Fencing2* niezależne rzutowanie punktów uniemożliwia poprawny odbiór sekwencji. Rzutowanie zapewniające ciągłość obiektów, nie dość, że pozwala na odwzorowanie sylwetki szermierza, umożliwia poprawną syntezę takich szczegółów, jak napisy na jego stroju.



Rysunek 7.8. Wirtualne zbliżenie się do sceny, sekwencja *Poznan_Fencing2* – synteza widoku z kamery skierowanej jak kamera 4, ale umieszczonej znacznie bliżej sceny; porównanie syntezy przy niezależnym rzutowaniu punktów (A, B) i rzutowaniu zapewniającym ciągłość obiektów (C, D)

W przypadku typowego scenariusza, gdy widz wirtualnie przemieszcza się pomiędzy rzeczywistymi widokami, rzutowanie zapewniające ciągłość obiektów również poprawia jakość syntezowanych widoków, jednakże poprawa ta nie jest wyraźna.

7.3.1. OBIEKTYWNY POMIAR JAKOŚCI

W celu pokazania wpływu zaproponowanej techniki na jakość syntezowanych widoków przeprowadzono eksperyment. Dla każdej sekwencji testowej przeprowadzono syntezę widoków wirtualnych przy zastosowaniu niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów.

Aby umożliwić obiektywny pomiar jakości, widoki wirtualne syntezowane były w pozycji widoków rzeczywistych (rysunek 3.1). Wyniki zaprezentowano w tabeli 7.1, przedstawiając wartości

Proponowana technika rzutowania punktów zapewniająca ciągłość obiektów

uśrednione dla wszystkich widoków (wartości dla poszczególnych widoków zaprezentowano w aneksie A3).

	PSNI	R [dB]	SSIM			
Sekwencja	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów		
BBB Butterfly	32,45	33,58	0,954	0,965		
BBB Flowers	25,68	25,97	0,872	0,878		
Poznan_Blocks	25,15	25,21	0,795	0,797		
Poznan_Blocks2	31,58	31,62	0,861	0,861		
Poznan_Fencing2	30,67	30,69	0,881	0,880		
Poznan_Service2	25,43	25,48	0,816	0,816		
Ballet	30,04	30,09	0,854	0,853		
Breakdancers	31,62	31,63	0,846	0,846		
Soccer Arc	21,14	21,21	0,756	0,755		
Soccer Linear	34,23	34,28	0,906	0,906		
Poznan_Carpark	33,85	33,88	0,932	0,932		
Poznan_Street	35,71	35,72	0,937	0,937		
Średnio	29,80	29,95	0,870	0,870		

Tabela 7.1. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów; wartości uśrednione dla wszystkich widoków

Dla każdej sekwencji testowej pogrubiono najlepszą otrzymaną jakość. Średnia poprawa jakości wyniosła 0,15 dB, przy czym zauważyć należy, iż dla syntetycznych sekwencji testowych, a więc sekwencji z poprawnymi mapami głębi, korzyść płynąca z rzutowania zapewniającego ciągłość obiektów jest większa – dla sekwencji *BBB Butterfly* wartość PSNR dla widoku wirtualnego zsyntezowanego przy użyciu rzutowania zapewniającego ciągłość obiektów jest o ponad 1 dB większa.

Zaprezentowane w tabeli 7.1 wyniki pokazują, iż zaproponowana technika rzutowania punktów z zachowaniem ciągłości obiektów poprawia jakość syntezowanych widoków, jednakże poprawa jakości nie jest duża. Przeprowadzony eksperyment nie pokazuje jednak różnicy jakości dla najbardziej typowego zastosowania opracowanej techniki, a więc wirtualnego zbliżenia się widza do sceny. Niestety, dla żadnej sekwencji testowej nie są dostępne widoki odniesienia dla takiego przypadku, a więc niemożliwa jest estymacja jakości obiektywnej tak syntezowanych widoków wirtualnych.

7.3.2. SUBIEKTYWNY POMIAR JAKOŚCI

W celu zmierzenia jakości w przypadku wirtualnego zbliżania się do sceny przeprowadzono testy subiektywne. W prezentowanych uczestnikom testów sekwencjach, wirtualna pozycja widza zmieniała się – w 10 sekund widz trzykrotnie przybliżał się do zarejestrowanej sceny, zaczynając od pozycji centralnej rzeczywistej kamery systemu wielokamerowego i poruszając się wzdłuż jej osi optycznej. Wynik przeprowadzonych testów zaprezentowano na rys. 7.9.



Rysunek 7.9. Poprawa mierzonej w sposób subiektywny jakości syntezowanych widoków przy zastosowaniu niezależnego rzutowania punktów i zaproponowanej techniki rzutowania zapewniającego ciągłość obiektów; dla każdej sekwencji oznaczono 95% przedział ufności; wartości dodatnie oznaczają, że jakość widoków syntezowanych przy użyciu MVS jest lepsza, niż w przypadku VSRS

Jak pokazano, zaproponowana technika rzutowania punktów zapewniająca ciągłość obiektów pozwala na znaczącą poprawę jakości subiektywnej syntezowanych widoków w przypadku wirtualnego zbliżania się do zarejestrowanej sceny. Dla wszystkich dwunastu sekwencji testowych osiągnięta poprawa jakości jest statystycznie istotna.

Proponowana technika rzutowania punktów zapewniająca ciągłość obiektów

8. PROPONOWANA TECHNIKA KOREKCJI NIESPÓJNOŚCI BARWNEJ WIDOKU WIRTUALNEGO

8.1. WPROWADZENIE

Jak zauważono w rozdziale 4.2, istotnym problemem w syntezie widoków wirtualnych dla rzadkich systemów wielokamerowych jest różna charakterystyka oświetlenia sceny w poszczególnych widokach rzeczywistych. Poza tym, różne kamery mogą cechować się różną charakterystyką barwną, a co za tym idzie w inny sposób rejestrować scenę. Oba wymienione problemy skutkują **niespójnością barwną**, a więc różnicą barwy tego samego punktu przerzutowanego z różnych widoków rzeczywistych. Niespójność barwna skutkuje gorszą jakością syntezowanych widoków.

Znane techniki korekcji barwnej [Fezza'14, Lu'15, Shao'08, Yamamoto'07, Ye'17] umożliwiają dopasowanie charakterystyki różnych widoków. Niemniej jednak, w celu zapewnienia widzowi wrażenia naturalności oglądanego materiału konieczne jest, aby charakterystyka barwna widoku wirtualnego dostosowana była do charakterystyki najbliższych widoków rzeczywistych. W przypadku dostosowania do widoków dalszych te obiekty, które odbijają światło w sposób nierównomierny, zostaną odwzorowane błędnie. Tym samym, chcąc odwzorować naturalne oświetlenie sceny, charakterystyka barwna powinna być dostosowywana do charakterystyki różnych widoków rzeczywistych – zależnie od tego, w którym miejscu znajduje się widz. Korekcja niespójności barwnej rzutowanych punktów musi być więc wykonywana **podczas syntezy**, a nie (jak w większości metod znanych z literatury) przed syntezą. Tym samym, zaproponowana technika powinna odznaczać się szybkością i niewielkim skomplikowaniem.

8.2. ADAPTACYJNA KOREKCJA NIESPÓJNOŚCI BARWNEJ

W przypadku opisywanej w rozprawie metody syntezy widoków wirtualnych korekcja niespójności barwnej [Dziembowski'18C] przeprowadzana jest podczas dwóch etapów syntezy (fioletowe bloki na rysunku 8.1):

- 1. Przy łączeniu pośrednich widoków wirtualnych wyznaczonych dla sąsiednich widoków rzeczywistych (punkt 8.2.1).
- 2. Przy wypełnianiu odsłonięć z dalszych widoków (punkt 8.2.2).

W odróżnieniu od technik opisywanych w rozdziale 2.5, gdzie korekcja barwna przeprowadzana była na widokach rzeczywistych, zaproponowana technika korekcji niespójności barwnej wykonywana jest po rzutowaniu punktów z widoków rzeczywistych do wirtualnego. Tym samym, nie jest konieczne wyszukiwanie odpowiadających sobie punktów, albowiem punkty

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

reprezentujące jeden punkt przestrzeni znajdują się we wszystkich pośrednich widokach wirtualnych w dokładnie tym samym miejscu.



Rysunek 8.1. Ogólny schemat proponowanej metody syntezy widoków wirtualnych. Schemat bloku przetwarzania pojedynczego widoku (PPW) przedstawiony został na rysunku 5.2

Jednym z problemów jakie musi rozwiązać korekcja niespójności barwnej, jest różnica barw tych samych obiektów w różnych widokach rzeczywistych spowodowana nielambertowskimi odbiciami światła od obiektów znajdujących się w rzeczywistej scenie. Przykładowe różnice barw tych samych obiektów zaprezentowano na rysunku 8.2.

Korekcja niespójności barwnej wspólna dla wszystkich punktów obrazu wprowadziłaby zniekształcenia, albo we wskazanych fragmentach, albo w pozostałej części widoku. Z tego powodu opracowano technikę adaptacyjną, która poprawia barwę poszczególnych punktów w zależności od ich otoczenia.



Rysunek 8.2. Problem nielambertowskich odbić: Poznan_Fencing2 - widok 1 (A) i 8 (C), Soccer Arc - widok 4 (B) i 7 (D)

8.2.1. KOREKCJA SĄSIEDNICH WIDOKÓW

W widoku wirtualnym połączonym z dwóch sąsiednich pośrednich widoków wirtualnych wyróżnić można cztery rodzaje punktów:

- 1. Punkty przerzutowane z obu widoków rzeczywistych;
- 2. Punkty przerzutowane wyłącznie z widoku lewego;
- 3. Punkty przerzutowane wyłącznie z widoku prawego;
- 4. Punkty nieprzerzutowane.

Punkty należące do pierwszej grupy charakteryzują się tym, iż ich barwa została wyznaczona poprzez zmieszanie barwy punktu przerzutowanego z lewego i prawego widoku rzeczywistego zgodnie z (5.7); barwa punktów z grupy drugiej i trzeciej natomiast została otrzymana poprzez skopiowanie barwy punktu z odpowiedniego pośredniego widoku wirtualnego.

Charakterystyka barwna punktów z pierwszej grupy płynnie zmienia się podczas zmiany punktu widzenia, a więc nie musi zostać poprawiona. Korekcji zostają poddane wyłącznie punkty należące do grupy drugiej i trzeciej.

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

Dla uproszczenia przedstawione zostanie przetwarzanie wyłącznie luminancji, jednakże wszystkie opisywane operacje wykonywane są w sposób niezależny dla wszystkich trzech składowych Y, Cb i Cr.

8.2.1.1. ALGORYTM KOREKCJI NIESPÓJNOŚCI BARWNEJ

W pierwszym kroku zarówno dla lewego jak i prawego widoku tworzona jest **mapa proporcji**. Sposób ich tworzenia przedstawiono w sposób schematyczny na rysunku 8.3. Dla każdego punktu z pierwszej grupy wyznaczany jest stosunek (proporcja) luminancji zmieszanej do luminancji przerzutowanej z pojedynczego widoku. Wartości tych proporcji umieszczane są w odpowiednich mapach proporcji:

$$M_L(x,y) = \frac{c_V(x,y)}{c_L(x,y)}, \qquad M_R(x,y) = \frac{c_V(x,y)}{c_R(x,y)}, \tag{8.1}$$

gdzie M_L i M_R są mapami proporcji dla lewego i prawego widoku, $c_L(x, y)$ i $c_R(x, y)$ luminancją przerzutowaną z lewego i prawego widoku a $c_V(x, y)$ zmieszaną luminancją w punkcie (x, y). W przypadku, gdy dany punkt nie został przerzutowany z obu widoków, stosunek nie jest liczony.



Rysunek 8.3. Tworzenie map proporcji

W drugim kroku następuje właściwa korekcja niespójności barwnej (rysunek 8.4). W kroku tym wszystkie punkty należące do grupy drugiej mnożone są przez lokalnie uśredniony współczynnik $\overline{M_L}(x, y)$, a grupy trzeciej przez $\overline{M_R}(x, y)$. Uśredniony współczynnik dla mapy proporcji lewego widoku wyznaczany jest ze wzoru:

$$\overline{M_L}(x,y) = \frac{1}{|\mathsf{P}|} \sum_{(w,h)\in\mathsf{P}} M_L(w,h) , \qquad (8.2)$$

gdzie P jest zbiorem wszystkich tych punktów w otoczeniu punktu (x, y), dla których znana jest wartość w mapie proporcji, a poprzez otoczenie rozumie się kwadratowe okno o wysokości i szerokości kilkunastu okresów próbkowania. Współczynnik $\overline{M_R}(x, y)$ liczony jest w sposób analogiczny.

Poprawiona wartość luminancji punktu widoku wirtualnego wyznaczana jest ze wzoru:

$$c_V(x, y) = c_L(x, y) \cdot M_L(x, y) \tag{8.3}$$

dla punktów przerzutowanych wyłącznie z lewego widoku oraz:

$$c_V(x,y) = c_R(x,y) \cdot \overline{M_R}(x,y)$$
(8.4)

dla punktów przerzutowanych z widoku prawego.



Rysunek 8.4. Algorytm adaptacyjnej korekcji niespójności barwnej: c_V – poprawiona wartość luminancji; c_L , c_R – wartość luminancji przerzutowana z lewego i prawego widoku; $\overline{M_L}$, $\overline{M_R}$ – uśredniony współczynnik dla lewej i prawej mapy proporcji

8.2.1.2. PRZYKŁAD DZIAŁANIA KOREKCJI NIESPÓJNOŚCI BARWNEJ

Sposób działania zaproponowanego algorytmu przedstawiono na przykładzie. Na rysunku 8.5A i 8.5B przedstawiono widoki pośrednie wyznaczone dla dwóch sąsiednich widoków rzeczywistych. Dla potrzeb przykładu założono, iż głębia wszystkich punktów jest identyczna, a kamera wirtualna umieszczona jest w połowie odległości między obiema kamerami rzeczywistymi. W takim przypadku luminancja punktów połączonego widoku wirtualnego wyznaczana z (5.7) jest średnią arytmetyczną luminancji w obu pośrednich widokach wirtualnych.

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

31		30	30	30	30		19	19			20	19		25	19	30	30	25	25
30	30	30		31	31		19	19		20	21	19		25	25	30	20	26	25
31	31	31			31		21	21		19	20	20		26	26	31	19	20	26
30	31	30	31		30		19			19	20	20		25	31	30	25	20	25
31	31	30	30		31		21		21	20	21	20		26	31	26	25	21	26
31	31	30	30		31		19		21	20	21	21		25	31	26	25	21	26
		1	1			_]	3			-			(2		

Rysunek 8.5. Łączenie pośrednich widoków wirtualnych bez korekcji niespójności barwnej (liczby reprezentują luminancję punktów); A – widok pośredni wyznaczony dla lewego widoku rzeczywistego, B – widok pośredni dla prawego widoku rzeczywistego, C – połączony widok wirtualny

Luminancja obiektu przerzutowanego z obu widoków rzeczywistych w sposób znaczny się różni, co skutkuje pojawieniem się błędów w widoku wirtualnym (rysunek 8.5C) w obszarach, które zostały przerzutowane wyłącznie z jednej kamery. Przykładowo, punkt oznaczony czerwoną ramką przerzutowany został wyłącznie z widoku lewego, a więc bez korekcji niespójności barwnej w połączonym widoku wirtualnym punkt ten miałby taką wartość luminancji widoku wirtualnego, jak w lewym widoku pośrednim, a więc 31.

W pierwszym kroku dla lewego i prawego pośredniego widoku wirtualnego wyznaczana jest mapa proporcji (M_L i M_R , wyznaczane zgodnie z (8.1)) – rysunek 8.6.

0,81				0,83	0,82		1,32				1,25	1,2
0,82	0,82			0,84	0,81		1,29	1,29			1,24	1,32
0,84	0,84				0,82		1,24	1,24				1,28
0,82			0,81		0,83		1,29			1,32		1,25
0,84		0,85	0,83		0,82		1,24		1,21	1,25		1,28
0,81		0,85	0,83		0,84]	1,32		1,21	1,25		1,24
			4			-			1	3		

Rysunek 8.6. Przykładowe mapy proporcji dla lewego (A) i prawego (B) pośredniego widoku wirtualnego; proporcje są wyznaczane wyłącznie dla punktów przerzutowanych z obu widoków

Następnie każdy punkt przerzutowany wyłącznie z jednego widoku przemnażany jest przez średnią wartość mapy proporcji w swoim otoczeniu. Przykładowo, dla zaznaczonego na rysunku 8.5A punktu liczona jest średnia wartość dostępnych próbek mapy proporcji w zaznaczonym oknie (dla czytelności przykładu mającym rozmiar 3×3) – wyniesie ona 0,82. Tym samym, wartość luminancji zaznaczonego punktu po korekcji niespójności barwnej wyniesie 25 ($31 \cdot 0,82 = 25$).

Na rysunku 8.7A i 8.7B pokazano poprawioną wartość luminancji punktów przerzutowanych z lewego i prawego widoku. Rysunek 8.7C przedstawia połączony widok wirtualny po korekcji niespójności barwnej.



Rysunek 8.7. Mapy proporcji dla lewego (A) i prawego (B) pośredniego widoku wirtualnego

W przypadku większych obszarów przerzutowanych z jednego tylko widoku zastosowane podejście byłoby jednak niewystarczające. Dla części punktów bowiem mogłoby się okazać, iż w ich otoczeniu mapa proporcji nie ma żadnych wartości (żaden punkt w otoczeniu punktu analizowanego nie został przerzutowany z obu widoków).

W celu zaradzenia temu problemowi przyjęto, by zaproponowany algorytm wykonywany był iteracyjnie, a wartości mapy proporcji były aktualizowane dla tych punktów (x, y), które w danej iteracji zostały poddane korekcji niespójności barwnej:

$$M_L(x,y) = \overline{M_L}(x,y), \qquad M_R(x,y) = \overline{M_R}(x,y), \qquad (8.5)$$

gdzie $\overline{M_L}(x, y)$ i $\overline{M_R}(x, y)$ oznaczają lokalnie uśrednione współczynniki (8.2) mapy proporcji, odpowiednio dla lewego i prawego pośredniego widoku wirtualnego.

8.2.2. KOREKCJA DALSZYCH WIDOKÓW

Korekcja niespójności barwnej dla punktów rzutowanych z dalszych widoków jest przeprowadzana w podobny sposób jak korekcja dla widoków sąsiednich, a więc poprzez stworzenie map proporcji. Korekcja dalszych widoków przeprowadzana jest jednak wyłącznie w obszarach nieprzerzutowanych z widoków sąsiednich.

Tak jak w przypadku korekcji sąsiednich widoków, dla każdego pośredniego widoku wirtualnego wyznaczane są trzy mapy proporcji – po jednej dla każdej składowej Y, Cb i Cr. Wszystkie składowe przetwarzane są w identyczny sposób, więc dla czytelności przedstawiona zostanie korekcja dla luminancji.

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

W pierwszym kroku dla każdego dalszego widoku *i* estymowana jest mapa proporcji:

$$M_i(x, y) = \frac{c_V(x, y)}{c_i(x, y)},$$
(8.6)

gdzie $c_V(x, y)$ jest wartością luminancji widoku wirtualnego (zawierającego informację z wszystkich pośrednich widoków wirtualnych), a $c_i(x, y)$ wartością luminancji w punkcie (x, y) *i*-tego pośredniego widoku wirtualnego. Wartość mapy proporcji wyznaczana jest wyłącznie dla tych punktów, dla których dostępne są wartości $c_V(x, y)$ i $c_i(x, y)$, i które znajdują się w sąsiedztwie obszarów nieprzerzutowanych z sąsiednich widoków.

W kolejnym kroku poprawiona wartość luminancji punktu (x, y) przerzutowanego z *i*-tego widoku rzeczywistego liczona jest poprzez wymnożenie poprzedniej wartości tegoż punktu przez uśredniony współczynnik $\overline{M_i}(x, y)$, wyznaczany analogicznie jak w (8.2).

W celu wyznaczenia współczynników korekcji dla większych obszarów widoku wirtualnego, zaproponowany algorytm jest algorytmem iteracyjnym, a wyznaczone uśrednione współczynniki mapy proporcji umieszczane są w tejże mapie:

$$M_i(x, y) = \overline{M_i}(x, y) . \tag{8.7}$$

8.3. SZYBKA KOREKCJA NIESPÓJNOŚCI BARWNEJ

Dodatkowy czas przetwarzania widoku wirtualnego potrzebny do przeprowadzenia opisanej w punkcie 8.5 adaptacyjnej korekcji niespójności barwnej wynosi – w zależności od sekwencji – od pół sekundy do około dwóch sekund na jedną ramkę. Czas ten jest znacząco krótszy, niż chociażby dla techniki opisanej w [Ye'17], jednakże wciąż zbyt długi, by zaproponowany algorytm mógł zostać użyty w systemie, w którym widoki wirtualne generowane są w czasie rzeczywistym, jak choćby w [Domański'18B].

Aby sprostać wymaganiom rzeczywistego systemu swobodnej nawigacji, a więc umożliwić syntezę czasu rzeczywistego, opracowano prostszy i znacznie szybszy algorytm korekcji niespójności barwnej [Dziembowski'18F]. Zauważono, iż najbardziej widocznymi błędami wynikającymi z różnej charakterystyki barwnej są przekłamania barwy na jednolitych powierzchniach – na tle. Zdecydowano się porzucić adaptacyjny charakter korekcji niespójności barwnej, starając się wyznaczyć globalną różnicę charakterystyk poszczególnych widoków.

Tak jak w przypadku opisanej wcześniej korekcji adaptacyjnej, szybka korekcja niespójności barwnej wykonywana jest dwukrotnie: podczas łączenia punktów przerzutowanych z sąsiednich widoków rzeczywistych i podczas wypełniania odsłonięć z pozostałych widoków.

W pierwszym kroku dla wszystkich tych punktów widoku wirtualnego, które zostały przerzutowane zarówno z lewego, jak i prawego widoku rzeczywistego (i których przerzutowana wartość glębi jest podobna), liczona jest różnica wszystkich trzech składowych Y, Cb i Cr pomiędzy barwą przerzutowaną z lewego widoku i barwą punktu z widoku prawego. Następnie wyznaczana jest uśredniona dla całego widoku wartość tej różnicy (ϵ), niezależnie dla każdej składowej. W ostatnim kroku średnia różnica dla poszczególnych składowych jest wykorzystywana do modyfikacji tych składowych dla punktów przerzutowanych z obu widoków rzeczywistych.

Poprawiona wartość poszczególnych składowych dla punktów przerzutowanych z widoku lewego dana jest wzorem:

$$c_L'(x,y) = c_L(x,y) - \frac{\epsilon \cdot d(L,V)}{d(L,V) + d(R,V)},$$
(8.8)

gdzie $c_L(x, y)$ jest wartością danej składowej w punkcie (x, y) przed korekcją niespójności barwnej, ϵ – średnią różnicą wartości danej składowej przerzutowanej z lewego i prawego widoku, a d(L, V)i d(R, V) są wyznaczane z (5.4) i stanowią odległość euklidesową pomiędzy kamerą wirtualną a, odpowiednio, lewą i prawą kamerą rzeczywistą.

Poprawiona wartość składowych barwnych dla punktów przerzutowanych z prawego widoku wyznaczana jest w sposób analogiczny, przy czym przed składnikiem odpowiadającym za poprawkę danej składowej znajduje się znak plus, a nie minus:

$$c'_{R}(x,y) = c_{R}(x,y) + \frac{\epsilon \cdot d(R,V)}{d(L,V) + d(R,V)}.$$
(8.9)

Podobne operacje wykonywane są w przypadku punktów przerzutowanych z dalszych widoków rzeczywistych. Dla każdego punktu połączonego widoku wirtualnego sprawdzane jest, czy został on przerzutowany również z dalszych widoków rzeczywistych. Jeżeli tak (i jeżeli przerzutowana wartość glębi jest podobna), dla tych widoków liczona jest różnica wszystkich trzech składowych jego barwy pomiędzy danym widokiem a połączonym widokiem wirtualnym. Następnie dla każdego widoku wyznaczana jest średnia wartość różnicy, niezależnie dla wszystkich trzech składowych.

Poprawiona wartość poszczególnych składowych barwnych dla punktów przerzutowanych z widoku rzeczywistego *i* dana jest wzorem:

$$c'_i(x, y) = c_i(x, y) - \epsilon_i , \qquad (8.10)$$

gdzie ϵ_i jest uśrednioną wartością różnicy danej składowej między widokiem *i* oraz połączonym widokiem wirtualnym.

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

Zaproponowane podejście nie wymaga przeprowadzania żadnych dodatkowych operacji, a do wykonania uproszczonej korekcji niespójności barwnej wymagane jest wykonanie jedynie dwóch dodatkowych operacji dodawania na każdy punkt obrazu dla każdego widoku rzeczywistego.

8.4. WYNIKI

Badania eksperymentalne przeprowadzono dla wszystkich 12 sekwencji testowych. Dla każdej z nich zbadano wpływ zaproponowanych technik korekcji niespójności barwnej na jakość syntezowanych widoków. Na rysunkach 8.8 – 8.11 przedstawiono wynik syntezy dla fragmentów czterech wybranych sekwencji. W każdym przypadku zamieszczono 4 obrazy: fragment widoku odniesienia, widoku zsyntezowanego bez korekcji niespójności barwnej, a także fragmenty dwóch widoków zsyntezowanych z użyciem zaproponowanych technik korekcji niespójności barwnej.

Jak pokazano, zaproponowana adaptacyjna technika korekcji umożliwia poprawę jakości subiektywnej, znacząco redukując widoczne w widoku wirtualnym błędy. Technika ta radzi sobie z problemem wzajemnego przysłaniania się obiektów w części widoków rzeczywistych, co w przypadku braku korekcji niespójności barwnej skutkuje pojawianiem się "cieni" bliższych obiektów na obiektach dalszych (rysunek 8.8C i 8.9C). Przy zastosowaniu adaptacyjnej korekcji niespójności barwnej obszary te są wypełniane w sposób poprawny (rysunki 8.8G i 8.9G). Podobnie w przypadku sekwencji *Soccer Arc*, gdzie widziana z różnych kamer murawa ma zupełnie inną charakterystykę barwną – zaproponowana korekcja pozwala usunąć widoczne na rysunku 8.10C "cienie" piłkarzy (rysunek 8.10G).



Rysunek 8.8. Sekwencja *Ballet*: fragmenty widoku odniesienia (A, B), widoku syntezowanego bez korekcji niespójności barwnej (C, D), z szybką korekcją niespójności barwnej (E, F) i z adaptacyjną korekcją niespójności barwnej (G, H); dla uwypuklenia różnic jasność i kontrast zwiększono o 40%

Technika umożliwia również korekcję dużo większych obszarów, jak choćby części pokazanego na rysunku 8.10D dużego fragmentu boiska. Po korekcji barwa murawy jest jednolita (rysunek 8.10H).



Rysunek 8.9. Sekwencja *Poznan_Fencing2*: fragmenty widoku odniesienia (A, B), widoku syntezowanego bez korekcji niespójności barwnej (C, D), z szybką korekcją niespójności barwnej (E, F) i z adaptacyjną korekcją niespójności barwnej (G, H); dla uwypuklenia różnic jasność zwiększono o 20%, a kontrast zmniejszono o 20%



Rysunek 8.10. Sekwencja *Soccer An*: fragment widoku odniesienia (A), widoku syntezowanego bez korekcji niespójności barwnej (B), z szybką korekcją niespójności barwnej (C) i z adaptacyjną korekcją niespójności barwnej (D); dla uwypuklenia różnic jasność i kontrast zwiększono o 40%

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

Zaproponowana technika adaptacyjna umożliwia przeprowadzenie korekcji niespójności barwnej nawet dla obszarów z niepoprawnie wyznaczoną glębią, jak choćby obszar podlogi na rysunku 8.8D czy pokazany na rysunku 8.9D obszar przy ręce szermierza (efekt korekcji na rysunkach 8.8H i 8.9H).

Ponadto, korekcja adaptacyjna umożliwia eliminację wpływu poświaty przy krawędziach obiektów. Przykładowo, obecna w widoku rzeczywistym poświata przy krawędzi kwiatu powoduje powstanie nieistniejących krawędzi w widoku wirtualnym (rysunek 8.11B). Korekcja niespójności barwnej umożliwia usunięcie tejże krawędzi (rysunek 8.11D), poprawiając subiektywną jakość syntezowanego widoku.



Rysunek 8.11. Sekwencja *BBB Flowers*: fragment widoku odniesienia (A), widoku syntezowanego bez korekcji niespójności barwnej (B), z szybką korekcją niespójności barwnej (C) i z adaptacyjną korekcją niespójności barwnej (D); dla uwypuklenia różnic kontrast zwiększono o 40%

Zaproponowany szybki algorytm korekcji niespójności barwnej nie pozwala poprawnie odtworzyć barwy obszarów o niepoprawnej głębi (rysunki 8.8F i 8.9F). Przy użyciu tej techniki nie jest również możliwa redukcja poświaty wokół obiektów (rysunek 8.11C).

Niemniej jednak, szybka korekcja umożliwia redukcję najbardziej widocznych błędów, a więc przekłamań barwy na jednolitym, nieruchomym tle (rysunki 8.8E i 8.9E). Również w przypadku sekwencji *Soccer Arc* błędy zostały znacząco zredukowane (rysunek 8.10E i 8.10F), choć w tym wypadku korekcja zadziałała gorzej, niż technika adaptacyjna.

Obie przedstawione techniki poprawiają inne elementy widoku wirtualnego: korekcja adaptacyjna dostosowuje lokalne charakterystyki obrazu, szybka korekcja natomiast dopasowuje globalną charakterystykę całych widoków. Zaproponowano więc połączenie obu przedstawionych technik, a więc przeprowadzanie korekcji adaptacyjnej na punktach wcześniej poprawionych szybką techniką korekcji.

8.4.1. OBIEKTYWNY POMIAR JAKOŚCI

W tabeli 8.1 przedstawiono jakość widoków wirtualnych syntezowanych bez zastosowania korekcji niespójności barwnej i w trzech konfiguracjach z włączoną korekcją niespójności barwnej.

Dla każdej sekwencji testowej pogrubiono największą otrzymaną wartość współczynnika PSNR. W przypadku sekwencji *Poznan_Carpark* i *Poznan_Street* brak poprawy jakości spowodowanej korekcją niespójności barwnej wynika z faktu, iż widoki z poszczególnych kamer zostały poddane wcześniejszej korekcji barwnej [Stankowski'10]. Analogiczną przyczynę wykazać można dla sekwencji *Soccer Linear*, która również – w ramach przetwarzania wstępnego – została poddana korekcji barwnej [Goorts'12]. Sekwencje *BBB Butterfly* i *BBB Flowers* są sekwencjami syntetycznymi, w których poszczególne widoki mają identyczną charakterystykę barwną.

	PSNR [dB]						
Sekwencja	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz			
BBB Butterfly	33,60	33,60	33,58	33,58			
BBB Flowers	25,99	25,99	25,97	25,97			
Poznan_Blocks	25,17	25,16	25,21	25,21			
Poznan_Blocks2	30,95	31,57	30,99	31,62			
Poznan_Fencing2	29,86	30,64	29,88	30,69			
Poznan_Service2	25,09	25,47	25,08	25,48			
Ballet	30,05	30,28	29,96	30,09			
Breakdancers	31,94	32,06	31,44	31,63			
Soccer Arc	20,87	21,02	21,26	21,21			
Soccer Linear	34,28	34,28	34,28	34,28			
Poznan_Carpark	33,90	33,90	33,88	33,88			
Poznan_Street	36,41	36,41	35,72	35,72			

Tabela 8.1. Jakość syntezowanych widoków dla czterech konfiguracji MVS, średnia dla wszystkich widoków

Jak pokazano, zaproponowane techniki korekcji niespójności barwnej zwiększają wartość PSNR dla tych naturalnych sekwencji wielowidokowych, które nie zostały poprawione barwnie w ramach przetwarzania wstępnego. Przedstawiona w tabeli poprawa jakości nie jest duża, jednakże zaprezentowane wartości zostały uśrednione dla wszystkich widoków. Analizując wartości PSNR dla pojedynczych widoków (tabele w aneksie A4) zaobserwować można jednak, iż dla niektórych widoków (na przykład czwartego i piątego widoku sekwencji *Poznan_Fencing2* czy czwartego widoku sekwencji *Ballet*) poprawa jakości spowodowana zastosowaniem korekcji niespójności barwnej jest dużo większa, sięgając odpowiednio 3 dB i 0,7 dB.

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

	SSIM						
Sekwencja	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz			
BBB Butterfly	0,965	0,965	0,965	0,965			
BBB Flowers	0,878	0,878	0,878	0,878			
Poznan_Blocks	0,794	0,794	0,797	0,797			
Poznan_Blocks2	0,859	0,860	0,860	0,861			
Poznan_Fencing2	0,874	0,879	0,877	0,880			
Poznan_Service2	0,812	0,815	0,814	0,816			
Ballet	0,856	0,856	0,853	0,853			
Breakdancers	0,856	0,857	0,845	0,846			
Soccer Arc	0,747	0,752	0,758	0,755			
Soccer Linear	0,906	0,906	0,906	0,906			
Poznan_Carpark	0,933	0,933	0,932	0,932			
Poznan_Street	0,938	0,938	0,937	0,937			

Tabela 8.2. Jakość syntezowanych widoków dla czterech konfiguracji MVS, średnia dla wszystkich widoków

Tabela 8.2 zawiera uśrednione dla wszystkich widoków i ramek wartości miary SSIM. Analizując rozkład pogrubionych wartości, a więc najlepszej jakości dla poszczególnych sekwencji, zauważyć można, iż wnioski wysnute z analizy tabeli 8.1 są prawidłowe – również w przypadku SSIM korekcja niespójności barwnej powoduje poprawę jakości w przypadku naturalnych sekwencji testowych.

8.4.2. SUBIEKTYWNY POMIAR JAKOŚCI

W celu pokazania wpływu zaproponowanej korekcji niespójności barwnej na subiektywną jakość syntezowanych widoków, przeprowadzono testy subiektywne. Podczas testów grupa uczestników porównywała jakość widoków syntezowanych bez korekcji niespójności barwnej z jakością widoków otrzymanych przy włączonych obu technikach korekcji. Wynik badań przedstawiono na rysunku 8.12.

Adrian Dziembowski, rozprawa doktorska



Rysunek 8.12. Poprawa mierzonej w sposób subiektywny jakości syntezowanych widoków z korekcją niespójności barwnej i bez korekcji; dla każdej sekwencji oznaczono 95% przedział ufności; wartości dodatnie oznaczają, że jakość widoków syntezowanych przy użyciu MVS jest lepsza, niż w przypadku VSRS

Jak zaprezentowano, dla większości sekwencji testowych poprawa jakości mierzonej w sposób subiektywny jest statystycznie istotna. Wyjątkiem są trzy sekwencje zarejestrowane systemem z liniowym rozmieszczeniem kamer, *Poznan_Carpark*, *Poznan_Street* i *Soccer Linear*, dla których jakość widoków syntezowanych z użyciem korekcji niespójności barwnej i bez niej jest porównywalna.

Największą poprawę jakości otrzymano dla sekwencji *Soccer Arc*, gdzie wyeliminowane zostały blędy pokazane na rysunku 8.10C i rysunek 8.10D, a także *Poznan_Fencing2* i *Ballet*, gdzie dopasowana została charakterystyka tych obszarów tla, które były widoczne wyłącznie w jednym sąsiednim widoku.

8.5. PODSUMOWANIE

W rozdziale 8. zaprezentowano dwie techniki korekcji niespójności barwnej punktów rzutowanych z różnych widoków rzeczywistych. W odróżnieniu od znanych z literatury technik korekcji barwnej umożliwiających korekcję barwną widoków rzeczywistych (przed syntezą), zaproponowane techniki umożliwiają korekcję niespójności barwnej rzutowanych punktów w czasie syntezy. Dzięki temu, możliwe jest odwzorowanie naturalnego oświetlenia sceny, a charakterystyka barwna widoku wirtualnego jest dopasowywana do wirtualnej pozycji (punktu widzenia) widza.

W ramach rozprawy zaproponowano adaptacyjną i szybką technikę korekcji niespójności barwnej. Szybka technika korekcji niespójności barwnej dopasowuje globalną charakterystykę barwną

Proponowana technika korekcji niespójności barwnej widoku wirtualnego

pośrednich widoków wirtualnych. W technice adaptacyjnej, niespójność barwna rzutowanego punktu jest poprawiana na podstawie jego otoczenia. Tym samym, technika ta umożliwia poprawną korekcję niespójności również dla sceny o bardzo zróżnicowanym oświetleniu. Przykładowe fragmenty widoków syntezowanych z użyciem zaproponowanych technik zaprezentowano na rysunkach 8.8 – 8.11.

Zaproponowane techniki korekcji niespójności barwnej umożliwiają niewielką poprawę jakości mierzonej w sposób obiektywny. Wartość PSNR dla widoków wirtualnych syntezowanych z użyciem obu opracowanych technik jest większa o 0,1 dB w porównaniu do widoków syntezowanych bez zastosowania korekcji. W przypadku miary SSIM różnica ta wynosi 0,001. Niemniej jednak, podstawowym zastosowaniem opracowanych technik jest eliminacja błędów barwnych widoku wirtualnego, które to przyciągają uwagę widza.

W przypadku subiektywnego pomiaru jakości, tylko dla 3 z 12 sekwencji testowych nie osiągnięto statystycznie istotnej poprawę jakości. Należy jednak podkreślić, iż wszystkie trzy sekwencje, dla których nie osiągnięto poprawy, zostały poddane wcześniejszej korekcji barwnej [Goorts'12, Stankowski'10]. Dla pozostałych 9 sekwencji testowych, poprawa jakości była statystycznie istotna. Co ciekawe, subiektywnie lepszą jakość otrzymano również dla sekwencji syntetycznych.

9. WSTĘPNE ZWIĘKSZANIE ROZDZIELCZOŚCI WIDOKÓW RZECZYWISTYCH I ODPOWIADAJĄCYCH IM MAP GŁĘBI

9.1. WPROWADZENIE

Kolejną techniką pozwalającą uzyskać lepszą jakość syntezowanych widoków jest wstępne zwiększanie rozdzielczości widoków rzeczywistych i odpowiadających im map głębi. Operacja ta wykonywana jest przed syntezą widoku wirtualnego.

Przy zastosowaniu takiego podejścia, synteza widoków wirtualnych przeprowadzana jest na obrazach zawierających większą liczbę punktów reprezentujących zarejestrowaną scenę, a tym samym zwiększona jest częstotliwość próbkowania.

Technika wstępnego zwiększania rozdzielczości jest znanym sposobem poprawy jakości syntezowanych widoków (np. [Senoh'17A, Zinger'10]), jak jednak zbadano w [Dziembowski'17B] zaproponowana przez autora prosta modyfikacja sposobu zwiększania rozdzielczości map głębi poprawia jakość syntezowanych widoków. W podejściu zaproponowanym przez autora rozprawy, sposób interpolacji brakujących próbek głębi różni się w zależności od obecności krawędzi obiektów w analizowanym fragmencie głębi. W przypadku, gdy analizowany fragment mapy głębi zawiera krawędź (duża różnica wartości głębi w sąsiednich próbkach), przeprowadzana jest interpolacja zerowego rzędu. Kiedy analizowany fragment mapy głębi jest gładki, brakujące próbki głębi interpolowane są liniowo.

Schemat zaproponowanej metody syntezy widoków wirtualnych z wstępnym zwiększaniem rozdzielczości przedstawiono na rysunku 9.1. Technika zwiększania rozdzielczości jest transparentna dla algorytmu syntezy – zmieniane są jedynie dane wejściowe i wyjściowe. Tym samym, jak pokazano w [Dziembowski'17A] i [Dziembowski'17B], może zostać ona zastosowana w dowolnym algorytmie syntezy bazującym na reprezentacji MVD.

W etapie zwiększania rozdzielczości modyfikowane są:

- 1. Widoki rzeczywiste;
- 2. Rzeczywiste mapy głębi;
- 3. Parametry wewnętrzne rzeczywistych kamer.

Parametry wewnętrzne poszczególnych kamer opisywane są macierzą **K** (1.1). Parametry odpowiadające dwukrotnie zwiększonym widokom i mapom glębi wyznaczane w następujący sposób:

Wstępne zwiększanie rozdzielczości widoków rzeczywistych i odpowiadających im map głębi

$$\mathbf{K_2} = \begin{bmatrix} 2 \cdot f \cdot s_x & f \cdot s_k & 2 \cdot o_x \\ 0 & 2 \cdot f \cdot s_y & 2 \cdot o_y \\ 0 & 0 & 1 \end{bmatrix}.$$
 (9.1)

Skośność przetwornika s_k nie ulega zmianie przy zwiększaniu rozdzielczości. Również parametry zewnętrzne kamer nie są modyfikowane, jako że zmianie nie uległo wzajemne ustawienie kamer, a jedynie ich wewnętrzna charakterystyka.



Rysunek 9.1. Schemat proponowanej metody syntezy widoków wirtualnych z wstępnym zwiększaniem rozdzielczości; fioletowy blok "2↑" odpowiada za dwukrotne zwiększenie rozdzielczości widoku rzeczywistego, mapy głębi i parametrów wewnętrznych kamery; blok "2↓" odpowiada za dwukrotne zmniejszenie rozdzielczości widoku wirtualnego; schemat bloku przetwarzania pojedynczego widoku (PPW) przedstawiony został na rysunku 5.2

9.2. Opis zaproponowanej techniki

W celu zwiększenia liczby próbek widoku rzeczywistego, stosowana jest prosta, biliniowa interpolacja brakujących punktów. W przypadku zwiększania rozdzielczości mapy glębi liniowa interpolacja próbek nie jest najlepszym rozwiązaniem. Na rysunku 9.2A przedstawiono fragment rzeczywistej mapy glębi. W przypadku liniowej interpolacji brakujących próbek zaznaczona pomarańczową linią krawędź pomiędzy obiektami zostanie rozmyta (rysunek 9.2C). W rozdziale 6.1.2 przedstawiono technikę poprawy rozmytych krawędzi glębi, jednakże jej użycie wiązałoby się z eliminacją części przerzutowanych próbek.

Drugą możliwością uzupełnienia brakujących próbek jest zastosowanie interpolacji zerowego rzędu, gdzie brakujące próbki są wypełniane wyłącznie na podstawie próbki wypełnionego punktu powyżej lub z lewej strony. W takim podejściu krawędź z oryginalnej mapy głębi zostanie zachowana (rysunek 9.2D), jednakże w przypadku gładkich, ciągłych obszarów wartości mapy głębi będą się zmieniać w sposób niepoprawny – skokowy.



Rysunek 9.2. Zwiększanie rozdzielczości rzeczywistej mapy glębi; mapa glębi: wejściowa (A), większej rozdzielczości (B), większej rozdzielczości z brakującymi próbkami wypełnianymi interpolacją pierwszego (C) i zerowego rzędu (D)

Biorąc pod uwagę zalety i wady obu sposobów interpolacji brakujących próbek, zaproponowano podejście mieszane, w którym wartość interpolowanych punktów powiększonej mapy głębi jest interpolowana liniowo w przypadku gładkich obszarów, a kopiowana z sąsiednich punktów w okolicach krawędzi obiektów. Wynik zaproponowanej interpolacji dla przykładu przedstawionego na rysunku 9.2 pokazano na rysunku 9.3B.



Rysunek 9.3. Powiększona mapa głębi: A – interpolacja liniowa, B – interpolacja mieszana, C – interpolacja 0. rzędu

135 / 266

Wstępne zwiększanie rozdzielczości widoków rzeczywistych i odpowiadających im map głębi

Kryterium wyboru sposobu interpolacji dla każdego interpolowanego punktu jest proste – jeżeli różnica głębi punktów, z których dany punkt miałby zostać zinterpolowany w sposób liniowy jest zbyt duża, wartość głębi w analizowanym punkcie zostanie skopiowana z próbki powyżej lub z lewej strony.

Powiększanie mapy glębi można by przeprowadzić również w bardziej wyrafinowany sposób, chociażby korzystając z segmentacji obrazu wraz z filtracją bilateralną [Dziembowski'16B], jednakże takie rozwiązanie znacząco wydłużyłoby czas obliczeń. Zaproponowany mieszany sposób interpolacji brakujących próbek pozwala na uzyskanie wartości PSNR większej o około 0,1 dB w porównaniu do interpolacji liniowej [Dziembowski'17B], lecz jest on otrzymywany praktycznie zerowym kosztem.

Zsyntezowany widok wirtualny charakteryzuje się dwukrotnie większą rozdzielczością, niż widoki rzeczywiste. Po syntezie dokonywane jest więc zmniejszenie rozdzielczości tegoż widoku, a dokonywane jest to poprzez biliniową decymację otrzymanego widoku wirtualnego. Tym samym rozmywane są krawędzie obiektów, dodając naturalności syntezowanemu widokowi wirtualnemu.

9.3. WYNIKI EKSPERYMENTALNE

Aby dokonać obiektywnego pomiaru jakości, widoki wirtualne syntezowane były w pozycji widoków rzeczywistych (rysunek 3.1). W tabelach 9.1 i 9.2 pokazano średnią (dla wszystkich widoków i ramek) jakość widoku wirtualnego. Wyniki dla poszczególnych widoków umieszczono w aneksie A5.

	PSNR [dB]						
Sekwencja	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości	Poprawa				
BBB Butterfly	32,98	33,58	0,60				
BBB Flowers	25,21	25,97	0,76				
Poznan_Blocks	25,02	25,21	0,19				
Poznan_Blocks2	31,00	31,62	0,62				
Poznan_Fencing2	30,10	30,69	0,59				
Poznan_Service2	25,17	25,48	0,31				
Ballet	29,66	30,09	0,43				
Breakdancers	31,29	31,63	0,34				
Soccer Arc	20,88	21,21	0,33				
Soccer Linear	34,18	34,28	0,10				
Poznan_Carpark	33,64	33,88	0,24				
Poznan_Street	35,49	35,72	0,23				
Średnio	29,55	29,95	0,40				

Tabela 9.1. Jakość syntezowanych widoków dla dwóch konfiguracji MVS, średnia dla wszystkich widoków

	SSIM						
Sekwencja	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości	Poprawa				
BBB Butterfly	0,956	0,965	0,009				
BBB Flowers	0,863	0,878	0,015				
Poznan_Blocks	0,786	0,797	0,011				
Poznan_Blocks2	0,858	0,861	0,003				
Poznan_Fencing2	0,873	0,880	0,007				
Poznan_Service2	0,808	0,816	0,008				
Ballet	0,841	0,853	0,012				
Breakdancers	0,829	0,846	0,017				
Soccer Arc	0,733	0,755	0,022				
Soccer Linear	0,906	0,906	0,000				
Poznan_Carpark	0,933	0,932	-0,001				
Poznan_Street	0,935	0,937	0,002				
Średnio	0,860	0,870	0,010				

Tabela 9.2. Jakość syntezowanych widoków dla dwóch konfiguracji MVS, średnia dla wszystkich widoków

Jak pokazano, wstępne zwiększanie rozdzielczości poprawia jakość syntezowanych widoków wirtualnych dla niemal wszystkich sekwencji testowych – zarówno dla PSNR, jak i SSIM.

Przedstawiona poprawa jakości została uzyskana kosztem większego nakładu obliczeń. Sam etap wstępnego zwiększania rozdzielczości nie jest skomplikowaną i czasochłonną operacją (dla każdej brakującej próbki powiększonego obrazu należy przepisać wartość z sąsiedniego punktu bądź wyznaczyć średnią maksymalnie czterech wartości), jednakże zwiększenie rozdzielczości wejściowych widoków i map głębi skutkuje przeprowadzaniem syntezy dla czterokrotnie większej liczby punktów, znacząco zwiększając czas potrzebny na zsyntezowanie widoku wirtualnego (tabela 9.3).

Wstępne zwiększanie rozdzielczości widoków rzeczywistych i odpowiadających im map głębi

	Czas syntezy 1 ramki [s]				
Sekwencja	Brak zwiększania	Wstępne zwiększanie			
	rozdzielczości	rozdzielczości			
BBB Butterfly	1,79	10,08			
BBB Flowers	2,00	11,02			
Poznan_Blocks	4,82	25,02			
Poznan_Blocks2	5,42	28,28			
Poznan_Fencing2	5,20	28,14			
Poznan_Service2	5,44	30,02			
Ballet	1,44	8,45			
Breakdancers	1,41	7,56			
Soccer Arc	4,09	24,43			
Soccer Linear	2,57	13,46			
Poznan_Carpark	3,49	17,55			
Poznan_Street	3,22	16,50			
Średnio	3,41	18,38			

Tabela 9.3. Jakość syntezowanych widoków dla dwóch konfiguracji MVS, średnia dla wszystkich widoków

10. Ocena efektywności zaproponowanej metody syntezy

10.1. WPROWADZENIE

We wcześniejszych rozdziałach opisano opracowaną przez autora rozprawy metodę syntezy (rozdział 5) z technikami poprawiającymi jakość widoków wirtualnych (rozdz. 6 – 9). Zaprezentowano wyniki syntezy widoków wirtualnych dotyczące zaproponowanych technik, porównując jakość widoków syntezowanych bez użycia i z użyciem danej techniki. Wszystkie zamieszczone wyniki dotyczyły jednak wyłącznie opracowanej metody MVS, dla której jedynie zmieniano konfigurację.

W niniejszym rozdziale zaproponowana metoda (MVS, *MultiView Synthesis*) została porównana z oprogramowaniem odniesienia (VSRS, *View Synthesis Reference Software*) [Senoh'17A] rozwijanym przez międzynarodową grupę ekspertów MPEG.

10.2. Ocena jakości syntezy

10.2.1. JAKOŚĆ MIERZONA W SPOSÓB OBIEKTYWNY

W tabeli 10.1 przedstawiono wartości PSNR syntezowanych widoków wirtualnych dla VSRS (lewa kolumna) i trzech konfiguracji zaproponowanej metody MVS.

	PSNR [dB]						
Sekwencja	WEDE	MVS	MVS	MVS			
	V 313	(2 widoki)	(4 widoki)	(wszystkie widoki)			
BBB Butterfly	31,25	32,52	33,37	33,58			
BBB Flowers	22,62	24,95	25,88	25,97			
Poznan_Blocks	24,78	25,25	25,21	25,21			
Poznan_Blocks2	29,60	29,89	31,53	31,62			
Poznan_Fencing2	29,62	30,51	30,69	30,69			
Poznan_Service2	24,68	25,09	25,50	25,48			
Ballet	29,21	30,09	30,10	30,10			
Breakdancers	30,62	31,62	31,64	31,64			
Soccer Arc	20,51	21,06	21,24	21,21			
Soccer Linear	34,18	34,03	34,26	34,28			
Poznan_Carpark	33,50	33,80	33,89	33,89			
Poznan_Street	35,48	35,69	35,72	35,72			
Średnio	28,84	29,54	29,92	29,95			

Tabela 10.1. Porównanie uśrednionej jakości syntezowanych widoków dla VSRS i MVS

Ocena efektywności zaproponowanej metody syntezy

W przypadku metody MVS jakość widoków wirtualnych estymowano dla przypadku użycia dwóch widoków rzeczywistych, czterech widoków rzeczywistych i wszystkich dostępnych widoków rzeczywistych.

Na rysunku 10.1 zaprezentowano poprawę jakości widoków syntezowanych przy użyciu opracowanej metody w porównaniu do oprogramowania odniesienia VSRS. Niebieskim kolorem oznaczono poprawę (względem VSRS) jakości widoków syntezowanych przy użyciu MVS w konfiguracji z wyłączonym wypełnianiem odsłonięć z dalszych widoków rzeczywistych, pomarańczowym – z wypełnianiem odsłonięć z dwóch dalszych kamer, a szarym MVS z użyciem informacji z wszystkich widoków rzeczywistych.



Rysunek 10.1. Poprawa jakości mierzonej jako PSNR w porównaniu do VSRS (wyniki dla trzech konfiguracji MVS: syntezy z użyciem dwóch, czterech i wszystkich dostępnych widoków rzeczywistych)

Jak przedstawiono, uśredniona dla wszystkich sekwencji wartość PSNR widoku syntezowanego MVS jest o ponad 1,1 dB większa, niż dla widoku generowanego przy użyciu VSRS.

Jeżeli wziąć pod uwagę wyłącznie sekwencje testowe zarejestrowane rzadkimi systemami wielokamerowymi (a więc wyłączając sekwencje *Soccer Linear, Poznan_Carpark* i *Poznan_Street*), korzyść

płynąca z zastosowania MVS jest jeszcze większa i wynosi niemal 1,5 dB (27,06 dB dla VSRS i 28,53 dB dla MVS).

Co ciekawe, w większości przypadków zwiększanie liczby widoków, z których wypelniane są odsłonięcia zmienia jakość syntezowanego widoku w sposób marginalny. Wyjątkiem jest sekwencja *BBB Butterfly*, dla której dosyntezowane z dalszych widoków tlo powoduje widoczną poprawę jakości.

W tabeli 10.2 przedstawiono porównanie jakości liczonej poprzez SSIM. Jak pokazano, średnio MVS pozwala osiągnąć o 0,015 lepszą jakość, niż VSRS.

Zauważyć należy, iż wartość SSIM dla MVS w konfiguracji z syntezą wyłącznie z dwóch i czterech widoków dla większości sekwencji jest niemalże identyczna, jak w przypadku syntezy ze wszystkich widoków rzeczywistych. Wyjątkiem jest *BBB Flowers*, która jest najbardziej skomplikowaną sekwencją, posiadając wiele wzajemnie przysłaniających się obiektów i dużą dynamikę głębi – tym samym dla *BBB Flowers* wiele jest obszarów odsłoniętych, które mogą zostać wypełnione informacją z innych widoków niż tylko dwa najbliższe.

	SSIM						
Sekwencja	VEDE	MVS	MVS	MVS (wszystkie widoki)			
	V 5K5	(2 widoki)	(4 widoki)				
BBB Butterfly	0,951	0,962	0,965	0,965			
BBB Flowers	0,824	0,870	0,877	0,878			
Poznan_Blocks	0,785	0,799	0,797	0,797			
Poznan_Blocks2	0,856	0,859	0,861	0,861			
Poznan_Fencing2	0,874	0,881	0,881	0,880			
Poznan_Service2	0,808	0,818	0,816	0,816			
Ballet	0,837	0,852	0,853	0,853			
Breakdancers	0,828	0,845	0,846	0,846			
Soccer Arc	0,714	0,755	0,753	0,755			
Soccer Linear	0,902	0,906	0,906	0,906			
Poznan_Carpark	0,929	0,933	0,932	0,932			
Poznan_Street	0,934	0,937	0,937	0,937			
Średnio	0,854	0,868	0,869	0,869			

Tabela 10.2. Porównanie uśrednionej jakości syntezowanych widoków dla VSRS i MVS

Dane zaprezentowane w tabeli 10.2 zobrazowano na rysunku 10.2. Na wykresie zaprezentowano poprawę (względem VSRS) jakości mierzonej jako SSIM dla trzech konfiguracji MVS. Kolorem niebieskim oznaczono poprawę jakości otrzymaną dla MVS z wyłączonym wypełnianiem

Ocena efektywności zaproponowanej metody syntezy

z dalszych widoków, pomarańczowym – dla syntezy z 4 widoków (2 sąsiednich i 2 dalszych), a szarym – dla syntezy z użyciem wszystkich widoków rzeczywistych.



Rysunek 10.2. Poprawa jakości mierzonej jako SSIM w porównaniu do VSRS (wyniki dla trzech konfiguracji MVS: syntezy z użyciem dwóch, czterech i wszystkich dostępnych widoków rzeczywistych)

Wyniki otrzymane dla poszczególnych widoków wszystkich sekwencji testowych zamieszczono w aneksie A6.

10.2.2. JAKOŚĆ MIERZONA W SPOSÓB SUBIEKTYWNY

W celu porównania mierzonej w subiektywny sposób jakości widoków wirtualnych syntezowanych przy użyciu zaproponowanej metody i metody odniesienia, przeprowadzono testy, w czasie których uczestnicy oceniali różnicę jakości widoków zsyntezowanych przy użyciu opracowanej w ramach rozprawy metody MVS i metody odniesienia VSRS. Czas trwania każdej oglądanej przez uczestników sekwencji wynosił 10 sekund.

Testy przeprowadzone zostały zgodnie z metodologią przedstawioną w rozdziale 3.6.2, a ich wyniki przedstawiono na rysunku 10.3.

Adrian Dziembowski, rozprawa doktorska



Rysunek 10.3. Poprawa mierzonej w sposób subiektywny jakości widoków syntezowanych MVS i VSRS; dla każdej sekwencji oznaczono 95% przedział ufności; wartości dodatnie oznaczają, że jakość widoków syntezowanych przy użyciu MVS jest lepsza, niż w przypadku VSRS

Jak pokazano, dla większości sekwencji testowych użycie zaproponowanej metody powoduje poprawę jakości mierzonej w sposób subiektywny. Dla dziewięciu z nich poprawa jakości jest statystycznie istotna. Wśród sekwencji, dla których nie osiągnięto statystycznie istotnej poprawy jakości są *Soccer Linear, Poznan_Carpark* i *Poznan_Street*, które to zostały zarejestrowane przy użyciu gęstych, liniowych systemów wielokamerowych. Zauważyć należy, iż użycie zaproponowanej metody nie spowodowalo statystycznie istotnej straty jakości dla żadnej sekwencji testowej, w tym sekwencji liniowych.

Największą poprawę jakości mierzonej subiektywnie osiągnięto dla obu sekwencji syntetycznych: *BBB Butterfly* i *BBB Flowers*. Jest to spowodowane faktem, iż obie sekwencje charakteryzują się większą dynamiką glębi, niż pozostałe sekwencje (zawierają zarówno bardzo bliskie obiekty, jak i obiekty znajdujące się w dużej odległości od kamer) – a przez to większą powierzchnią odsłonięć, niż pozostałe sekwencje (tabela 4.2). Jednocześnie, dla sekwencji *BBB Flowers* i *BBB Butterfly* dostępne są perfekcyjne mapy glębi, co pozwala na poprawne przerzutowanie obiektów z dalszych widoków rzeczywistych.

W przypadku pozostałych sekwencji zarejestrowanych przy użyciu rzadkich systemów wielokamerowych z nieliniowo rozmieszczonymi kamerami, poprawa jakości jest mniejsza, lecz również statystycznie istotna.

Ocena efektywności zaproponowanej metody syntezy

10.2.3. JAKOŚĆ SYNTEZY PODCZAS WIRTUALNEJ NAWIGACJI

We wszystkich wcześniejszych testach w przypadku sekwencji syntetycznych, a więc BBB Butterfly i BBB Flowers używano siedmiu widoków rzeczywistych. Dla obu sekwencji dostępnych jest jednak aż 79 widoków rzeczywistych.

Przeprowadzono eksperyment, w którym syntezę przeprowadzano dla 7 wybranych widoków rzeczywistych, pozostałe 72 widoki traktując jako widoki odniesienia w celu wyznaczenia jakości syntezy. Tym samym możliwe było zbadanie jakości syntezy w przypadku płynnej zmiany punktu widzenia, czyli wirtualnego przemieszczania się w scenie.

Na rysunkach 10.4 i 10.5 przedstawiono PSNR wyznaczony dla widoków wirtualnych zsyntezowanych w pozycjach 72 widoków odniesienia, odpowiednio dla sekwencji *BBB Butterfly* i *BBB Flowers*. Linią niebieską oznaczono wyniki otrzymane przy użyciu oprogramowania odniesienia, linia pomarańczowa odpowiada wynikom dla opracowanej metody bez wstępnego zwiększania rozdzielczości, linia szara – MVS z dwukrotnym zwiększaniem rozdzielczości.



Rysunek 10.4. Porównanie PSNR [dB] syntezowanych widoków dla VSRS i MVS dla wszystkich widoków sekwencji BBB Butterfly, wartość uśredniona dla wszystkich 120 ramek sekwencji


Rysunek 10.5. Porównanie PSNR [dB] syntezowanych widoków dla VSRS i MVS dla wszystkich widoków sekwencji *BBB Flowers*, wartość uśredniona dla wszystkich 120 ramek sekwencji

W przypadku sekwencji *BBB Butterfly*, MVS z dwukrotnym zwiększaniem rozdzielczości zapewnia lepszą jakość syntezy niż VSRS dla dowolnego punktu widzenia (dla dowolnego widoku wirtualnego). Konfiguracja bez wstępnego zwiększania rozdzielczości zapewnia lepszą jakość niż VSRS dla niemal wszystkich widoków wirtualnych. Jedyny wyjątek stanowią widoki 18, 72 i 73, a więc widoki z wirtualnych kamer znajdujących się w bezpośrednim otoczeniu kamer rzeczywistych, a największa strata jakości wprowadzana przez MVS sięga 0,4 dB.

Średnia różnica jakości dla wszystkich widoków wynosi niemal 1,7 dB, a największa różnica 6,1 dB na korzyść zaproponowanej metody. W przypadku wstępnego zwiększania rozdzielczości wartości te dodatkowo rosną odpowiednio do niemal 2,1 dB i 6,6 dB.

W przypadku sekwencji *BBB Flowers*, dla wszystkich punktów widzenia lepsza jakość syntezy zostanie uzyskana przy użyciu metody MVS niezależnie od tego, czy widoki i mapy głębi zostaną wstępnie zwiększone, czy też nie. Średni zysk z zastosowania zaproponowanej metody wyniósł ponad 2,4 dB, w skrajnym przypadku osiągając 5,3 dB w przypadku braku wstępnego powiększenia widoków i map głębi oraz 2,75 dB i 5,5 dB dla widoków wstępnie zwiększonych.

Tabele z dokładnymi wartościami PSNR i SSIM wyznaczonymi dla poszczególnych widoków wirtualnych zamieszczono w aneksie A7.

Ocena efektywności zaproponowanej metody syntezy

Osiągnięte rezultaty pozwalają stwierdzić, iż zaproponowana metoda jest efektywniejsza, niż algorytm opisany w [Ceulemans'18], dla którego poprawa jakości (w porównaniu do VSRS) dla sekwencji *BBB Butterfly* wyniosła średnio 1,22 dB i maksymalnie 4,77 dB, a dla sekwencji *BBB Flowers* odpowiednio 1,47 dB i 3,99 dB.

10.3. POMIAR CZASU OBLICZEŃ

Oprócz jakości syntezowanych widoków, istotny jest czas syntezy pojedynczego widoku wirtualnego. W celu zbadania i porównania czasu obliczeń przeprowadzono eksperyment. Komputer, jakiego użyto do testów miał następującą charakterystykę: procesor Intel i7-4770 (3,4 GHz), 32 GB pamięci RAM, dysk SSD.

W tabeli 10.3 przedstawiono porównanie czasu obliczeń dla oprogramowania VSRS oraz MVS w przypadku syntezy z użyciem dwóch, czterech i wszystkich widoków rzeczywistych. Zaprezentowany w tabeli czas jest czasem potrzebnym do zsyntezowania widoku wirtualnego w najlepszej konfiguracji zaproponowanego algorytmu, a więc z włączonym rzutowaniem wymuszającym ciągłość obiektów, korekcją niespójności barwnej (adaptacyjną i szybką), a także wstępnym zwiększaniem rozdzielczości.

	Czas syntezy 1 ramki [s]						
Sekwencja	VEDE	MVS	MVS	MVS			
	V 5K5	(2 widoki)	(4 widoki)	(wszystkie widoki)			
BBB Butterfly	1,58	6,15	7,94	10,08			
BBB Flowers	1,54	4,75	7,89	11,02			
Poznan_Blocks	3,68	9,51	14,03	25,02			
Poznan_Blocks2	3,64	13,99	22,31	28,28			
Poznan_Fencing2	2,24	11,44	18,79	28,14			
Poznan_Service2	2,42	12,01	18,48	30,02			
Ballet	0,90	2,80	4,57	8,45			
Breakdancers	0,86	2,72	5,08	7,56			
Soccer Arc	2,62	14,22	16,75	24,43			
Soccer Linear	1,74	4,52	8,90	13,46			
Poznan_Carpark	2,10	6,70	12,35	17,55			
Poznan_Street	2,16	6,85	11,84	16,50			
Średnio	2,12	7,97	12,41	18,38			

Tabela 10.3. Czas potrzebny do wygenerowania jednego widoku wirtualnego – dla VSRS i 3 konfiguracji MVS (dla syntezy z użyciem informacji z 2 i 4 sąsiednich widoków rzeczywistych oraz wszystkich widoków rzeczywistych (z wstępnym zwiększaniem rozdzielczości))

Jak pokazano, średni czas potrzebny do zsyntezowania jednej ramki sekwencji jest zdecydowanie większy, niż dla oprogramowania odniesienia – nawet w przypadku syntezy wyłącznie z dwóch sąsiednich widoków rzeczywistych. Głównym czynnikiem zwiększającym czas syntezy jest zastosowanie wstępnego zwiększania rozdzielczości. Gdy technika ta jest bowiem używana, wszystkie operacje wykonywane są na czterokrotnie większym obrazie.

Co prawda technika wstępnego zwiększania rozdzielczości pozwala zwiększyć jakość syntezowanych widoków, jednakże czyni ona proces syntezy zdecydowanie dłuższym. Dlatego też, w tabeli 10.4 przedstawiono czas potrzebny do wygenerowania ramki bez wstępnego zwiększania rozdzielczości.

	Czas syntezy 1 ramki [s]							
Sekwencja	VSDS	MVS	MVS	MVS				
	V SKS	(2 widoki)	(4 widoki)	(wszystkie widoki)				
BBB Butterfly	1,58	1,06	1,41	1,79				
BBB Flowers	1,54	0,92	1,44	2,00				
Poznan_Blocks	3,68	1,63	2,65	4,82				
Poznan_Blocks2	3,64	2,74	3,98	5,42				
Poznan_Fencing2	2,24	2,00	3,17	5,20				
Poznan_Service2	2,42	2,25	3,09	5,44				
Ballet	0,90	0,51	0,81	1,44				
Breakdancers	0,86	0,51	0,87	1,41				
Soccer Arc	2,62	2,46	3,14	4,09				
Soccer Linear	1,74	0,89	1,60	2,57				
Poznan_Carpark	2,10	1,27	2,33	3,49				
Poznan_Street	2,16	1,19	2,19	3,22				
Średnio	2,12	1,45	2,22	3,41				

Tabela 10.4. Czas potrzebny do wygenerowania jednego widoku wirtualnego – dla VSRS i 3 konfiguracji MVS (dla syntezy z użyciem informacji z 2 i 4 sąsiednich widoków rzeczywistych oraz wszystkich widoków rzeczywistych (bez wstępnego zwiększania rozdzielczości))

Dzięki wyłączeniu wspomnianej techniki czas syntezy zmniejszył się około czterokrotnie. W przypadku użycia 4 lub mniejszej liczby widoków rzeczywistych czas syntezy dla MVS stał się porównywalny z VSRS. Wykorzystanie wszystkich dostępnych widoków rzeczywistych zwiększa czas potrzebny na zsyntezowanie 1 widoku wirtualnego do niecałych 3,5 sekundy.

Trzy i pół sekundy na ramkę nie jest jednak czasem, który umożliwiałby syntezę widoków wirtualnych w rzeczywistym systemie swobodnego punktu widzenia. Aby umożliwić widzowi

Ocena efektywności zaproponowanej metody syntezy

swobodną nawigację w scenie, synteza widoków wirtualnych musi działać w czasie rzeczywistym [Domański'18C]. Z tego względu w ramach pracy przeprowadzono również badania opisane w rozdziale 11.

11. SYNTEZA CZASU RZECZYWISTEGO

11.1. ANALIZA CZASU OBLICZEŃ

W celu opracowania metody syntezy widoków wirtualnych działającej w czasie rzeczywistym (rtMVS, *real-time MultiView Synthesis*), zbadano czas potrzebny na wykonanie każdego etapu algorytmu. Wyniki przedstawiono w tabeli 11.1 i na rysunku 11.1.

Etap syntezy widoku wirtualnego	Czas obliczeń [s]	Czas obliczeń [% całego czasu]
Rzutowanie mapy głębi	0,28	8
Filtracja wirtualnej mapy głębi	0,26	8
Rzutowanie widoku rzeczywistego	0,27	8
Rzutowanie zapewniające ciągłość	0,75	22
Filtracja widoku wirtualnego	0,13	4
Szybka korekcja niespójności barwnej	0,17	5
Adaptacyjna korekcja niespójności barwnej	0,98	29
Łączenie sąsiednich widoków pośrednich	0,15	4
Wypełnianie odsłonięć	0,13	4
Uzupełnianie widoku wirtualnego	0,28	8
Cały proces syntezy	3,41	100

Tabela 11.1. Średni czas wykonania poszczególnych etapów syntezy widoku wirtualnego



Rysunek 11.1. Procentowy udział czasu wykonania poszczególnych etapów syntezy widoku wirtualnego

Synteza czasu rzeczywistego

Najbardziej czasochlonnymi operacjami są adaptacyjna korekcja niespójności barwnej i rzutowanie zapewniające ciągłość obiektów. Łącznie, te dwie operacje zajmują ponad połowę czasu syntezy pojedynczego widoku. Obie wymienione techniki umożliwiają poprawę jakości syntezowanych widoków wirtualnych, jednakże tworząc metodę syntezy działającą w czasie rzeczywistym należy znaleźć kompromis pomiędzy jakością, a szybkością działania. Z tego powodu, w szybkiej metodzie syntezy widoków wirtualnych dokonywana będzie szybka korekcja niespójności barwnej, a punkty będą rzutowane w sposób niezależny. W tabeli 11.2 przedstawiono rozkład czasu potrzebnego na poszczególne etapy syntezy po wyłączeniu dwóch najbardziej czasochłonnych operacji.

Etan avetary widely, wistuelness	Cras obligacó [s]	Czas obliczeń
Etap syntezy włoku writuaniego	Czas obliczen [s]	[% całego czasu]
Rzutowanie mapy glębi	0,29	17
Filtracja wirtualnej mapy glębi	0,25	15
Rzutowanie widoku rzeczywistego	0,27	16
Filtracja widoku wirtualnego	0,13	8
Szybka korekcja niespójności barwnej	0,16	10
Łączenie sąsiednich widoków pośrednich	0,17	10
Wypełnianie odsłonięć	0,13	8
Uzupełnianie widoku wirtualnego	0,27	16
Cały proces syntezy	1,68	100

Tabela 11.2. Średni czas wykonania poszczególnych etapów syntezy widoku wirtualnego

Jak zaprezentowano, czas potrzebny na wykonanie poszczególnych etapów algorytmu po wyłączeniu adaptacyjnej korekcji niespójności barwnej i techniki rzutowania zapewniającego ciągłość obiektów nie zmienił się.

Zgodnie z wynikami zamieszczonymi w tabeli 10.1, jakość widoków syntezowanych przy użyciu czterech widoków nie różni się znacząco od jakości widoków syntezowanych z wszystkich widoków rzeczywistych. Aby dodatkowo przyspieszyć działanie syntezy, ograniczono więc liczbę używanych widoków do czterech: dwóch sąsiednich oraz dwóch dalszych (do wypełniania odsłonięć). Czas przetwarzania dla takiego scenariusza zaprezentowano w tabeli 11.3.

Dla przedstawionej konfiguracji zmniejszył się (w porównaniu do syntezy z wszystkich widoków) czas tych operacji, które wykonywane są dla punktów przerzutowanych z wszystkich widoków rzeczywistych, a więc etapów wypisanych w pierwszych pięciu wierszach tabeli 11.3. Czas łączenia sąsiednich pośrednich widoków wirtualnych pozostał bez zmian, jako że same sąsiednie

widoki wciąż są używane. Skróceniu uległo wypełnianie odsłonięć informacją z dalszych widoków – tych oczywiście jest mniej. Jedyną operacją, jakiej czas wzrósł, jest uzupełnianie syntezowanego widoku. Z powodu mniejszej liczby użytych widoków rzeczywistych mniej punktów widoku wirtualnego zostanie wypełnione informacją z dalszych widoków. Tym samym, większy obszar widoku wirtualnego musi zostać uzupełniony.

Etap syntezy widoku wirtualnego	Czas obliczeń [s]	Czas obliczeń [% całego czasu]
Rzutowanie mapy głębi	0,14	12
Filtracja wirtualnej mapy głębi	0,13	11
Rzutowanie widoku rzeczywistego	0,14	12
Filtracja widoku wirtualnego	0,07	6
Szybka korekcja niespójności barwnej	0,08	6
Łączenie sąsiednich widoków pośrednich	0,17	14
Wypełnianie odsłonięć	0,09	8
Uzupełnianie widoku wirtualnego	0,35	30
Cały proces syntezy	1,16	100

Tabela 11.3. Średni czas wykonania poszczególnych etapów syntezy widoku wirtualnego

Na tym etapie algorytm średnio wymaga 1,16 sekundy przetwarzania do wygenerowania jednego widoku wirtualnego. Aby widoki syntezowane były w czasie rzeczywistym, jedna ramka powinna być generowana w maksymalnie 40 milisekund (dla 25 obrazów na sekundę), a więc niemal 29 razy szybciej.

11.2. REDUKCJA CZASU OBLICZEŃ

W celu osiągnięcia wymaganej szybkości obliczeń dokonano szeregu optymalizacji oraz uproszczeń. Podstawowymi zmianami są:

- 1. Zastąpienie typowej syntezy wstecz metodą hybrydową.
- 2. Uproszczenie filtracji widoku i mapy głębi.
- 3. Uproszczenie sposobu łączenia informacji przerzutowanej z różnych widoków.
- 4. Uproszczenie algorytmu uzupełniania niezsyntezowanych obszarów.

Jak napisano w rozdziale 2.2, w syntezie wstecz rzutowanie punktów przeprowadzane jest dwa razy – pierwszy raz w celu wyznaczenia wirtualnej mapy głębi i drugi raz w celu przerzutowania punktów z widoków rzeczywistych na bazie tejże mapy. Rzutowanie punktów wykonywane jest poprzez czasochlonne mnożenie pozycji i glębi tych punktów przez macierze projekcji, zgodnie z (1.3).

W celu uniknięcia drugiego cyklu rzutowania, opracowano szybką, hybrydową metodę syntezy [Dziembowski'18A, Dziembowski'18D]. Zmodyfikowany schemat przetwarzania pojedynczego widoku (PPW) przedstawiono na rysunku 11.2.



Rysunek 11.2. Zmodyfikowany schemat przetwarzania informacji z pojedynczego widoku rzeczywistego wraz z odpowiadającą mu mapą glębi (schemat przed modyfikacjami przedstawiono na rysunku 5.2)

W metodzie tej rzutowanie punktów odbywa się wyłącznie raz – w celu wyznaczenia wirtualnej mapy glębi. Jest to możliwe dzięki temu, iż dla każdego punktu wirtualnej mapy glębi zapamiętywana jest pozycja, z której został on przerzutowany do widoku wirtualnego. Przykładowo, jeżeli punkt znajdujący się w wirtualnej mapie glębi na pozycji (x_V, y_V) znajdował się na pozycji (x_R, y_R) w widoku rzeczywistym, położenie (x_R, y_R) zostanie dla niego zapamiętane.

W drugim kroku, barwa poszczególnych punktów widoku wirtualnego mogłaby być wprost skopiowana z odpowiednich zapamiętanych pozycji. Takie podejście jednak byłoby tradycyjną, choć inaczej zaimplementowaną, syntezą w przód. Jak jednak napisano, opracowana metoda jest metodą hybrydową, łącząc zalety syntezy w przód (szybkość działania) z zaletami syntezy wstecz, umożliwiając filtrację wirtualnej mapy glębi.

Filtracja wirtualnej mapy głębi dokonywana jest w dwójnasób. Po pierwsze, brakujące próbki głębi są interpolowane na podstawie próbek przerzutowanych. Równolegle dla wypełnianych próbek głębi (rysunek 11.3A, 11.3C) interpolowane są również pozycje odpowiadających im punktów w widoku rzeczywistym (rysunek 11.3B, 11.3D). Pozycje te wyznaczane są z dokładnością do **połowy**

okresu próbkowania – dzięki temu operacja ta może zostać wykonana poprzez przesunięcie bitowe. W przypadku, gdy pozycja odpowiadającego punktu w widoku rzeczywistym nie jest liczbą całkowitą, do punktu widoku wirtualnego kopiowana jest uśredniona barwa dwóch (w przypadku jednej niecałkowitej współrzędnej) bądź czterech (gdy obie współrzędne są niecałkowite) najbliższych punktów widoku rzeczywistego.

30		30	30	30	30	30	30	30	30	30	30
30	30	30		29	29	30	30	30	29.5	29	29
30	30	30			30	30	30	30	30	30	30
29	29	29	29		30	29	29	29	29	29.5	30
28	28	28	28		30	28	28	28	28	29	30
28	28	28	28		30	28	28	28	28	29	30
		I	ł					(2		
(3, 2)		(5, 2)	(6, 3)	(7, 3)	(8, 3)	(3, 2)	(4, 2)	(5, 2)	(6, 3)	(7,3)	(8, 3)
(3, 3)	(4, 3)	(5, 3)		(6, 4)	(7, 4)	(3, 3)	(4, 3)	(5, 3)	(5.5, 3.5)	(6, 4)	(7,4)
(3, 4)	(4, 4)	(5, 4)	(6, 4)		(8, 5)	(3, 4)	(4, 4)	(5, 4)	(6, 4)	(7, 4.5)	(8, 5)
(2, 5)	(3, 5)	(4, 5)	(5, 5)		(8, 6)	(2, 5)	(3, 5)	(4, 5)	(5, 5)	(6.5, 5.5)	(8, 6)
(1,6)	(2, 6)	(3, 6)	(4, 6)		(8,7)	(1, 6)	(2, 6)	(3, 6)	(4, 6)	(6, 6.5)	(8,7)
(1,7)	(2, 7)	(3, 7)	(4, 7)		(9, 8)	(1,7)	(2, 7)	(3, 7)	(4, 7)	(6.5, 7.5)	(9, 8)
		τ	2					Г	`		

Rysunek 11.3. Fragment wirtualnej mapy głębi przed (A) i po filtracji (C) wraz z tablicą pozycji odpowiadających punktów w widoku rzeczywistym przed (B) i po filtracji (D)

Również filtracja widoku wirtualnego została uproszczona. W rozdziale 6.2 opisano technikę bazującą na gradiencie otoczenia. W realizacji czasu rzeczywistego technika ta została zmodyfikowana. W uproszczonym algorytmie dla każdego punktu znajdującego się przy krawędziach obiektów w mapie głębi sprawdzane jest, czy ma on większą głębię niż punkt po jego prawej stronie (punkty otoczone czerwoną ramką na rysunku 11.4A). Jeżeli tak, porównywana jest luminancja analizowanego punktu z luminancją punktu po stronie lewej (punkty otoczone czerwoną ramką na rysunku 11.4B). W przypadku znacznej różnicy punkt nie jest rzutowany do widoku wirtualnego (punkty oznaczone czerwoną czcionką na rysunku 11.4B). Jeżeli to punkt po lewej stronie charakteryzował się większą

Synteza czasu rzeczywistego

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	5	5	10	10
10	10	10	5	5	5	10	10
10	10	5	5	5	5	10	10
10	10	5	5	5	5	10	10
10	10	5	5	5	5	10	10
A							

głębią niż punkt analizowany, różnica luminancji sprawdzana jest dla lewego punktu i punktu z nim sąsiadującego.

Rysunek 11.4. Fragment rzeczywistej mapy glębi (A) i odpowiadającego jej widoku rzeczywistego (B); w obu przypadkach czerwoną ramką oznaczono pary porównywanych punktów, a czerwoną czcionką punkty nierzutowane

Analogiczne przetwarzanie wykonywane jest również w kierunku pionowym. Tym samym, dla każdego punktu widoku rzeczywistego leżącego przy krawędzi głębi wykonywane są jedynie cztery operacje porównania.

W rozdziale 5.3 przedstawiono sposób łączenia informacji przerzutowanej z dwóch widoków, bazujący na mieszaniu barwy punktów zależnie od odpowiadającej im wartości wirtualnej mapy glębi i odległości pomiędzy kamerą wirtualną i obiema sąsiednimi kamerami rzeczywistymi. W metodzie czasu rzeczywistego również ten etap został uproszczony, a mieszanie barw realizowane jest poprzez średnią arytmetyczną barw przerzutowanych z obu widoków, a więc może być zaimplementowane poprzez przesunięcia bitowe.

Analizując tabelę 11.3 zauważono, iż zaproponowana technika uzupełniania zajmuje aż 30% czasu całej syntezy widoku wirtualnego. Tym samym, w syntezie czasu rzeczywistego zastosowano zdecydowanie szybszą, acz mniej dokładną technikę uzupełniania widoku. W zastosowanym algorytmie uzupełniania wartość wypełnianych punktów wyznaczana jest na podstawie wartości czterech punktów – najbliższych przerzutowanych punktów z lewej i z prawej strony, a także najbliższych punktów powyżej i poniżej (rysunek 11.5A).

W pierwszym kroku sprawdzane jest, który z czterech analizowanych punktów ma największą głębię. Następnie dla trzech pozostałych punktów sprawdzane jest, czy ich głębia jest podobna do wyznaczonego maksimum. Jeżeli tak, punkty są uwzględniane przy uzupełnianiu – tym samym do uzupełniania może zostać użyte od jednego do czterech punktów.

Barwa uzupełnianego punktu jest wyznaczana na jeden z czterech sposobów:

- 1. Jest kopiowana w przypadku, gdy największą głębię ma tylko 1 analizowany punkt (obszar zaznaczony na rysunku 11.5B na żółto).
- 2. Jest uśredniana z wszystkich 4 punktów, jeżeli wszystkie mają podobną głębię.
- 3. Jest uśredniana z dwóch punktów, jeżeli 2 analizowane punkty miały podobną głębię (obszar zaznaczony na zielono).
- 4. Gdy 3 punkty miały podobną głębię (obszar niebieski), barwa uzupełnianego punktu jest uśredniana z czterech wartości – wartości dwóch przeciwległych punktów (lewego i prawego albo powyżej i poniżej) i dwukrotnie powtórzonej wartości trzeciego punktu.

27 27

39 26

39

40 40

39 40

39

39 41

39 40

38

40





Dzięki takiemu podejściu możliwe jest wykonanie uzupełniania z użyciem operacji przesunięcia bitowego i sumy, bez używania czasochlonnej operacji dzielenia.

Znalezienie najbliższych przerzutowanych punktów po czterech stronach wszystkich uzupelnianych punktów zaimplementowano w sposób wymagający wykonania wyłącznie dwóch pętli po wszystkich punktach widoku wirtualnego – w pierwszej pętli dla wszystkich uzupelnianych punktów wyznaczone są pozycje najbliższego lewego i górnego przerzutowanego punktu, w pętli drugiej – pozycje punktu poniżej i po prawej stronie.

Czas potrzebny na wykonanie poszczególnych operacji dla przyspieszonej wersji algorytmu syntezy widoków wirtualnych przedstawiono w tabeli 11.4.

Synteza czasu rzeczywistego

Etap syntezy widoku wirtualnego	Czas obliczeń [ms]	Czas obliczeń [% całego czasu]
Rzutowanie mapy głębi	50,00	57
Filtracja wirtualnej mapy głębi	6,40	7
Rzutowanie widoku rzeczywistego	8,26	9
Filtracja widoku wirtualnego	6,41	7
Szybka korekcja niespójności barwnej	6,00	7
Łączenie sąsiednich widoków pośrednich	1,04	1
Wypełnianie odsłonięć	5,00	6
Uzupełnianie widoku wirtualnego	4,17	5
Cały proces syntezy	87,28	100

Tabela 11.4. Średni czas wykonania poszczególnych etapów syntezy widoku wirtualnego (dla sekwencji o rozdzielczości 1920×1080)

Dzięki zastosowanym uproszczeniom i optymalizacjom udało się zmniejszyć czas obliczeń z 1160 ms do niecałych 90 ms. Zauważyć należy jednak, iż podany czas wyznaczony został dla implementacji jednowątkowej. W przypadku użycia większej liczby wątków czas ten oczywiście się zmniejszy.

Najprostszą do zrównoleglenia operacją jest rzutowanie mapy głębi. Operacja ta jest wykonywana niezależnie dla czterech używanych rzeczywistych map głębi, a jej wynikiem są cztery osobne mapy głębi, odpowiadające poszczególnym pośrednim widokom wirtualnym. Tym samym operacja może zostać w pełni zrównoleglona, a jej czas wykonania zmaleje czterokrotnie, skutkując czasem syntezy jednego widoku mniejszym niż 50 ms.

Czas przetwarzania jednej ramki sekwencji wielowidokowej wysokiej rozdzielczości wynoszący 50 ms oznacza, iż przedstawiony algorytm umożliwia syntezę czasu rzeczywistego dla częstotliwości 20 obrazów na sekundę.

11.3. OCENA JAKOŚCI SYNTEZY

Po dokonaniu optymalizacji przeprowadzono badania sprawdzające, czy redukcja czasu obliczeń nie wpłynęła w sposób istotny na jakość syntezowanych widoków. W tabelach 11.5 i 11.6 jakość widoków syntezowanych w czasie rzeczywistym (za pomocą opracowanej metody rtMVS – *real-time MultiView Synthesis*) porównano z jakością widoków syntezowanych przy użyciu MVS i oprogramowania odniesienia VSRS. Szczegółowe wyniki zamieszczono w aneksie A8.

Solavoncia		PSNR [dB]				
Serwencja	VSRS	rtMVS	MVS			
BBB Butterfly	31,25	32,39	32,98			
BBB Flowers	22,62	24,76	25,21			
Poznan_Blocks	24,78	24,91	25,02			
Poznan_Blocks2	29,60	30,30	31,00			
Poznan_Fencing2	29,62	29,68	30,10			
Poznan_Service2	24,68	24,80	25,17			
Ballet	29,21	29,26	29,66			
Breakdancers	30,62	31,28	31,29			
Soccer Arc	20,51	20,69	20,88			
Soccer Linear	34,18	34,14	34,18			
Poznan_Carpark	33,50	33,62	33,64			
Poznan_Street	35,48	35,48	35,49			
Średnio	28,84	29,28	29,55			

Tabela 11.5. Jakość syntezowanych widoków dla VSRS, rtMVS, MVS, średnia dla wszystkich widoków (w przypadku MVS użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela 11.6. Jakość syntezowanych widoków dla VSRS, rtMVS, MVS, średnia dla wszystkich widoków (w przypadku MVS użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

S al much aire	SSIM					
Serwencja	VSRS	rtMVS	MVS			
BBB Butterfly	0,951	0,954	0,956			
BBB Flowers	0,824	0,856	0,863			
Poznan_Blocks	0,785	0,779	0,786			
Poznan_Blocks2	0,856	0,857	0,858			
Poznan_Fencing2	0,874	0,871	0,873			
Poznan_Service2	0,808	0,803	0,808			
Ballet	0,837	0,834	0,841			
Breakdancers	0,828	0,828	0,829			
Soccer Arc	0,714	0,720	0,733			
Soccer Linear	0,902	0,905	0,906			
Poznan_Carpark	0,929	0,931	0,933			
Poznan_Street	0,934	0,935	0,935			
Średnio	0,854	0,856	0,860			

Jak zaprezentowano, jakość widoków syntezowanych przy użyciu zaproponowanej metody czasu rzeczywistego jest gorsza, niż w przypadku widoków syntezowanych algorytmem MVS. Taki wynik był jednak wynikiem oczekiwanym, jako że w rtMVS zastosowano uproszczenia zmniejszające jakość syntezowanych widoków. Pomimo tego, widoki syntezowane przy użyciu rtMVS mają wyraźnie lepszą jakość, niż widoki syntezowane za pomocą oprogramowania odniesienia. Jednocześnie, czas estymacji pojedynczego widoku został mocno zredukowany. Średni czas estymacji dla wszystkich sekwencji testowych i badanych metod porównano w tabeli 11.7.

	Czas syntezy 1 ramki [ms]						
Sekwencja	VSRS	rtMVS (1 wątek)	rtMVS (4 wątki)	MVS			
BBB Butterfly	1275	42	24	887			
BBB Flowers	1488	47	28	877			
Poznan_Blocks	3764	97	54	1700			
Poznan_Blocks2	2738	89	50	1443			
Poznan_Fencing2	2219	89	49	2119			
Poznan_Service2	2189	91	52	1984			
Ballet	989	34	19	504			
Breakdancers	745	34	21	532			
Soccer Arc	2379	90	49	2150			
Soccer Linear	1524	56	33	1074			
Poznan_Carpark	2151	74	40	1577			
Poznan_Street	1883	74	42	1501			
Średnio	1945	68	38	1362			

Tabela 11.7. Średni czas syntezy pojedynczego widoku dla VSRS, rtMVS (implementacja 1- i 4-wątkowa) i MVS (w przypadku MVS użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

W przypadku implementacji czterowątkowej, jedynym równolegle wykonywanym etapem było rzutowanie mapy głębi. Pozostałe operacje wykonywane były na jednym wątku procesora.

Co istotne, w odróżnieniu od innych implementacji czasu rzeczywistego, realizowanych na FPGA [Akin'15, Wang'12] czy kartach graficznych [Do'11, Rogmans'09, Yao'16], przedstawiony algorytm zaimplementowano na **procesorze**. Tym samym wykazano, iż możliwa jest szybka synteza widoków wirtualnych nawet dla tanich komputerów czy laptopów ze zintegrowanymi kartami graficznymi.

11.4. PODSUMOWANIE

W rozdziale 11.1 przedstawiono sposób redukcji czasu obliczeń dla zaproponowanej metody syntezy widoków wirtualnych. Jak pokazano w tabeli 11.7, nawet dla jednowątkowej implementacji zaproponowanej metody rtMVS możliwe jest uzyskanie syntezy czasu rzeczywistego dla sekwencji mniejszych rozdzielczości.

W przypadku implementacji czterowątkowej, możliwa jest synteza czasu rzeczywistego również dla sekwencji wyższej rozdzielczości, przy czym w przypadku sekwencji o rozdzielczości 1920×1080 otrzymywana częstotliwość oscyluje wokół 20 syntezowanych obrazów na sekundę. Tym niemniej wykazano, iż możliwe jest stworzenie efektywnej metody syntezy widoków wirtualnych wykorzystującej informację z dalszych widoków rzeczywistych.

Jak zaprezentowano w tabeli 11.5, jakość widoków syntezowanych z użyciem zaprezentowanej metody czasu rzeczywistego (rtMVS) jest wyższa, niż w przypadku oprogramowania odniesienia (VSRS) lecz niższa, niż dla opracowanej w ramach rozprawy metody MVS. Należy zauważyć jednak, iż metoda rtMVS powstała poprzez dokonanie szeregu uproszczeń w metodzie MVS. Poprawa jakości widoków wirtualnych syntezowanych przy użyciu szybkiej metody syntezy rtMVS i zbliżenie się do jakości widoków syntezowanych za pomocą MVS stanowić będzie temat dalszych prac badawczych.

Synteza czasu rzeczywistego

12. PODSUMOWANIE

12.1. UZASADNIENIE TEZY ROZPRAWY

W rozdziale 1.4.1 sformułowano tezę rozprawy:

TEZA

Możliwe jest opracowanie wydajnej metody syntezy widoków wirtualnych dobrej jakości poprzez wykorzystanie dalszych widoków rzeczywistych, filtrację krawędzi w widokach rzeczywistych i odpowiadających im mapach glębi oraz korekcję niespójności barwnej widoku wirtualnego.

W celu sprawdzenia prawdziwości tezy przeprowadzono szereg badań, w których wykorzystano autorską metodę syntezy widoków wirtualnych opracowaną w ramach rozprawy. Poszczególne fragmenty tezy wykazano badając poszczególne opracowane techniki, dotyczące kolejno: wykorzystania dalszych widoków rzeczywistych, filtracji i korekcji niespójności barwnej. Przebieg tych badań opisano w rozdziałach 4 – 11. Wszystkie rezultaty uzyskano z wykorzystaniem zbioru 12 sekwencji testowych.

W rozdziale 4. dokonano porównania jakości widoków wirtualnych, w których do wypełnienia odsłonięć użyto punktów z dalszych widoków rzeczywistych z jakością widoków wirtualnych, w których obszary odsłonięte były uzupełniane na podstawie otoczenia. Średnia poprawa jakości spowodowana użyciem informacji z dalszych widoków wyniosła **0,41 dB**.

Aby uniezależnić otrzymane rezultaty od wpływu zaproponowanego algorytmu uzupełniania dokonano również porównania z oprogramowaniem odniesienia. Jako, że w oprogramowaniu odniesienia nie przewidziano możliwości wypełniania odsłonięć z innych widoków, porównania dokonano wyłącznie dla obszarów wypełnionych z pozostałych widoków rzeczywistych. Średni wzrost jakości wynikający z zastosowania wypełniania odsłonięć wyniósł ponad **2 dB**.

W rozdziale 6. opisano zaproponowane algorytmy filtracji krawędzi obiektów w widokach rzeczywistych (rozdział 6.2) i odpowiadających im mapach głębi (rozdział 6.1.2). Jakość widoków syntezowanych przy zastosowaniu opracowanych technik filtracji porównano z jakością widoków otrzymanych bez przeprowadzenia dodatkowej filtracji krawędzi.

Dla przyjętego zbioru sekwencji testowych zaproponowana technika filtracji krawędzi umożliwiła uzyskanie średnio **0,1 dB** większej wartości PSNR, niż w przypadku braku dodatkowej filtracji. Zdecydowanie większą poprawę jakości uzyskano w przypadku oceny jakości subiektywnej,

Podsumowanie

gdzie dla 9 spośród 12 sekwencji testowych (w tym wszystkich sekwencji zarejestrowanych za pomocą rzadkich systemów wielokamerowych) otrzymano statystycznie istotną poprawę jakości.

W 8. rozdziale rozprawy opisano dwie opracowane techniki korekcji niespójności barwnej:

- 1. Technikę adaptacyjną, pozwalającą dopasować lokalną charakterystykę barwną obszarów widoku wirtualnego przerzutowanych z różnych widoków rzeczywistych,
- 2. Technikę globalną, wyrównującą charakterystykę barwną wszystkich punktów rzutowanych z poszczególnych widoków rzeczywistych.

Jak pokazano w rozdziale 8, zaproponowane techniki umożliwiają zwiększenie jakości syntezowanych widoków wirtualnych. W przypadku zastosowania obu technik korekcji niespójności barwnej średni wzrost jakości mierzonej poprzez PSNR wyniósł niemal **0,2 dB**. Tak jak w przypadku filtracji, również korekcja niespójności barwnej ma większy wpływ na jakość mierzoną w sposób subiektywny. Jak zaprezentowano na rysunku 8.12, dla 8 z 12 sekwencji testowych osiągnięto statystycznie istotną różnicę jakości.

W rozdziale 11. pokazano, iż zaprezentowany algorytm może – po dokonaniu kilku uproszczeń i optymalizacji – działać w czasie rzeczywistym, również dla sekwencji wysokiej rozdzielczości. Opracowana metoda rtMVS pozwoliła na uzyskanie wartości PSNR większą średnio o **0,5 dB** w porównaniu do widoków syntezowanych przy użyciu oprogramowania odniesienia przy jednoczesnym **51-krotnym** skróceniu czasu obliczeń.

Jak pokazano w rozdziale 11, dla implementacji jednowątkowej uzyskano średni czas syntezy jednego widoku mniejszy niż 70 ms. W przypadku użycia czterech wątków czas ten zmniejszył się do niecałych **40 ms** w przypadku uśrednienia czasu syntezy dla wszystkich sekwencji i niecałych **50 ms** dla sekwencji o rozmiarze obrazu 1920×1080.

12.2. PRZEPROWADZONE BADANIA

W celu przeprowadzenia badań zawartych w rozprawie autor stworzył ponad **13 000 linii kodu**, wliczając w to ponad 11 500 linii napisanych w języku C++ i 1500 linii w języku Python. Poza tym, w celu przeprowadzenia porównań z oprogramowaniem odniesienia VSRS autor dokonywał modyfikacji w tym oprogramowaniu (liczącym niemal 6000 linii kodu w języku C++). Tym samym, autor rozprawy pracował z niemal 20 tysiącami linii kodu.

Autor uczestniczył również w rejestracji i kalibracji wielowidokowych sekwencji testowych, wliczając w to sekwencje *Poznan_Blocks*, *Poznan_Blocks2* i *Poznan_Fencing2*.

Eksperymenty przeprowadzone w celu wygenerowania wyników zawartych w rozprawie wymagały niemal **200 godzin** nieprzerwanych obliczeń na jednym wątku komputera.

12.3. Oryginalne osiągnięcia

Najistotniejszym oryginalnym osiągnięciem rozprawy jest opracowanie dostosowanej do systemów z rzadkim rozmieszczeniem kamer metody syntezy widoków wirtualnych, w której do syntezy widoku wirtualnego wykorzystywana jest informacja pochodząca z wielu widoków rzeczywistych. W opracowanej metodzie obszary nieprzerzutowane z dwóch sąsiednich widoków rzeczywistych wypełniane są z widoków dalszych.

Analizując rezultaty dla dwunastu zróżnicowanych sekwencji testowych wykazano, iż jakość widoków wirtualnych syntezowanych w zaproponowany sposób jest lepsza, niż jakość widoków syntezowanych przy użyciu oprogramowania odniesienia. Średnia poprawa wartości PSNR wyniosła niemal **1,2 dB**. Co ważne, dla wszystkich sekwencji zarejestrowanych przy użyciu rzadkich systemów wielokamerowych uzyskano **statystycznie isotną poprawę** jakości mierzonej w sposób subiektywny.

Ponadto, opracowano kilka technik poprawiających jakość syntezowanych widoków:

- 1. Technikę filtracji rozmytych krawędzi w widokach rzeczywistych bazującą na analizie gradientów,
- 2. Technikę filtracji rozmytych krawędzi w rzeczywistych mapach glębi,
- 3. Wymuszającą ciągłość obiektów technikę rzutowania punktów z widoków rzeczywistych,
- 4. Technikę adaptacyjnej korekcji niespójności barwnej, wyrównującą lokalną charakterystykę obszarów przerzutowanych z różnych widoków rzeczywistych,
- 5. Technikę szybkiej korekcji niespójności barwnej, wyrównującą globalną, wspólną dla całych widoków charakterystykę barwną obszarów rzutowanych z różnych widoków rzeczywistych,
- 6. Technikę wstępnego zwiększania rozdzielczości widoków rzeczywistych i odpowiadających im map glębi.

Niewątpliwym osiągnięciem rozprawy jest również opracowanie algorytmu syntezy widoków wirtualnych działającego w **czasie rzeczywistym** bez konieczności zastosowania kart graficznych czy programowalnych układów FPGA. Co istotne, zaproponowany algorytm umożliwia syntezę widoków mających średnio o **0,5 dB** większą wartość PSNR niż widoki syntezowane przy użyciu oprogramowania odniesienia VSRS przy jednoczesnym **51-krotnym** skróceniu czasu obliczeń.

Podsumowując, w rozprawie wykazano, iż jest możliwe stworzenie efektywnej metody syntezy widoków wirtualnych dla rzadkich systemów wielokamerowych, a tym samym możliwe jest stworzenie systemu swobodnej nawigacji, w którym użytkownik może wirtualnie poruszać się w scenie zarejestrowanej przy pomocy rzadkiego systemu wielokamerowego.

Podsumowanie

13. **BIBLIOGRAFIA**

Pogrubiono publikacje, których autorem lub współautorem jest autor rozprawy.

[Ahn'13]	I. Ahn, C. Kim, <i>A novel depth-based virtual view synthesis method for free viewpoint video</i> , IEEE Transactions on Broadcasting, tom 59, s. 614-626, 2013
[Akin'15]	A. Akin, R. Capoccia, J. Narinx, J. Masur, A. Schmid, Y. Leblebici, <i>Real-time free viewpoint synthesis using three-camera disparity estimation hardware</i> , IEEE International Symposium on Circuits and Systems, ISCAS 2015, Lizbona, Portugalia, 24-27.05.2015
[AVC]	ISO/IEC, Coding of audio-visual objects, Part 10: Advanced Video Coding, ISO/IEC IS 14496-10, 2014
[Barnes'09]	C. Barnes, E. Shechtman, A. Finkelstein, D.B. Goldman, <i>Patch-Match:</i> <i>a randomized correspondence algorithm for structural image editing</i> , ACM Transactions on Graphics – TOG, tom 28, nr 3, 08.2009
[Bay'08]	H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, <i>Speeded-Up Robust Features</i> (SURF), Computer Vision and Image Understanding, tom 110, nr 3, s. 346-359, 06.2008
[Bertalmio'00]	M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, Image inpainting, SIGGRAPH 2000, Nowy Orlean, Stany Zjednoczone, 23-28.07.2000
[Bertalmio'03]	M. Bertalmio, L. Vese, G. Sapiro, S. Osher, <i>Simultaneous structure and texture image inpainting</i> , IEEE Transactions on Image Processing, tom 12, nr 8, 09.2003
[Bielicki'13]	J. Bielicki, R. Sitnik, <i>A method of 3D object recognition and localization in a cloud of points</i> , EURASIP Journal on Advances in Signal Processing, tom 2013, s. 1-13, 2013
[Bonatto'17]	D. Bonatto, A. Schenkel, T. Lenertz, Y. Li, G. Lafruit, ULB high density 2D/3D camera array data set, version 2, ISO/IEC JTC1/SC29/WG11 MPEG2017, M40806, Turyn, Włochy, 17-21.07.2017
[Bondarev'11]	E. Bondarev, R. Miquel, M. Imbert, S. Zinger, P.H.N. de With, On the technology roadmap of free-viewpoint 3DTV receivers, IEEE International Conference on Consumer Electronics, ICCE 2011, Las Vegas, Stany Zjednoczone, 9-12.01.2011
[Buyssens'15]	P. Buyssens, M. Daisy, D. Tschumperle, O. Lezoray, Depth-aware patch- based image disocclusion for virtual view synthesis, SIGGRAPH Asia, Kobe, Japonia, 2-6.11.2015
[Buyssens'17]	P. Buyssens, O. Le Meur, M. Daisy, D. Tschumperlé, Olivier Lézoray, Depth-guided disocclusion inpainting of synthesized RGB-D images, IEEE Transactions on Image Processing, tom 26, nr 2, s. 525-538, 02.2017

[Ceulemans'16]	B. Ceulemans, S.P. Lu, G. Lafruit, A. Munteanu, <i>Efficient MRF-based disocclusion inpainting in multiview video</i> , IEEE International Conference on Multimedia and Expo, ICME 2016, Seattle, Stany Zjednoczone, 11-15.07.2016
[Ceulemans'18]	B. Ceulemans, S.P. Lu, G. Lafruit, A. Munteanu, <i>Robust multiview synthesis for wide-baseline camera arrays</i> , IEEE Transactions on Multimedia, tom 20, nr 9, s. 2235-2248, 09.2018
[Chen'06]	G. Chen, Y. Liu, N. Max, Real-time view synthesis from a sparse set of views, Signal Processing: Image Communication, tom 22, nr 2, s. 188-202, 02.2007
[Cho'17]	J.H. Cho, W. Song, H. Choi, T. Kim, <i>Hole filling method for depth image based rendering based on boundary decision</i> , IEEE Signal Processing Letters, tom 24, nr 3, s. 329-333, 03.2017
[Cignoni'08]	P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, G. Ranzuglia, <i>Meshlab: an open-source mesh processing tool</i> , 6th Eurographics Italian Chapter Conference, Salerno, Włochy, 2-4.07.2008
[Cooke'06]	E. Cooke, P. Kauff, T. Sikora, <i>Multi-view synthesis: a novel view creation approach for free viewpoint video</i> , Signal Processing: Image Communication, tom 21, nr 6, s. 476-492, 06.2006
[Criminisi'04]	A. Criminisi, P. Prez, K. Toyama, Region filling and object removal by exemplar- based image inpainting, IEEE Transactions on Image Processing, tom 13, nr 9, s. 1200-1212, 2004
[Cyganek'02]	B. Cyganek, <i>Komputerowe przetwarzanie obrazów trójwymiarowych</i> , Akademicka Oficyna Wydawnicza EXIT, 2002
[Cyganek'09]	B. Cyganek, S.J. Paul, An introduction to 3D computer vision techniques and algorithms, Wiley, 2009
[Daribo'10]	I. Daribo, B. Pesquet-Popescu, <i>Depth-aided image inpainting for novel view synthesis</i> , IEEE International Workshop on Multimedia Signal Processing, MMSP 2010, Saint-Malo, Francja, 4-6.10.2010
[Debevec'96]	P.E. Debevec, C.J. Taylor, J. Malik, <i>Modeling and rendering architecture from photographs: a hybrid geometry and image-based approach</i> , 23rd annual conference on Computer graphics and interactive techniques, Nowy Orlean, Stany Zjednoczone, 4-9.08.1996
[Do'11]	L. Do, G. Bravo, S. Zinger, P.H.N. de With, <i>Real-time free-viewpoint DIBR</i> on GPUs for large base-line multi-view 3DTV videos, Visual Communications and Image Processing, VCIP 2011, Tainan, Tajwan, 6-9.11.2011
[Domański'09A]	M. Domański, M. Gotfryd, K. Wegner, View synthesis for multiview video transmission, International Conference on Image Processing, Computer Vision and Pattern Recognition, IPCV 2009, Las Vegas, Stany Zjednoczone, 13-16.07.2009

	Adrian Dziembowski, rozprawa doktorska
[Domański'09B]	M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, K. Wegner, <i>Poznań multiview video test sequences and camera parameters</i> , ISO/IEC JTC1/SC29/WG11 MPEG2009, M17050, Xi'an, Chiny, 26-30.10.2009
[Domański'11]	M. Domański, Obraz cyfrowy, Wydawnictwa Komunikacji i Łączności, 2010
[Domański'13]	M. Domański, O. Stankiewicz, K. Wegner, M. Kurc, J. Konieczny, J. Siast, J. Stankowski, R. Ratajczak, T. Grajek, <i>High efficiency 3D video coding using new tools based on view synthesis</i> , IEEE Transactions on Image Processing, tom 22, s. 3517-3527, 2013
[Domański'14A]	M. Domański, A. Dziembowski, A. Kuehn, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, <i>Experiments on acquisition and processing of video for free-viewpoint television</i> , 3DTV Conference 2014, Budapeszt, Węgry, 2-4.07.2014
[Domański'14B]	M. Domański, D. Mieloch, A. Dziembowski, O. Stankiewicz, K. Wegner, Super multiview image compression: results for Bee sequence (FTV EE3), ISO/IEC JTC1/SC29/WG11 MPEG2014, M35070, Strasburg, Francja, 20- 24.10.2014
[Domański'14C]	M. Domański, A. Dziembowski, A. Kuehn, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, <i>Poznan Blocks – a multiview video test sequence and camera parameters for free viewpoint television</i> , ISO/IEC JTC1/SC29/WG11 MPEG2014, M32243, San Jose, Stany Zjednoczone, 13-17.01.2014
[Domański'14D]	M. Domański, A. Dziembowski, A. Kuehn, D. Mieloch, Telewizja swobodnego punktu widzenia – nowa usługa czy futurystyczna wizja?, Przegląd Telekomunikacyjny, nr 8-9, s. 734-737, 2014
[Domański'15A]	M. Domański, A. Dziembowski, D. Mieloch, A. Łuczak, O. Stankiewicz, K. Wegner, <i>A practical approach to acquisition and processing of free viewpoint video</i> , 31st Picture Coding Symposium, PCS 2015, Cairns, Australia, 31.05-3.06.2015
[Domański'15B]	M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, Ł. Kowalski, M. Kurc, A. Łuczak, D. Mieloch, R. Ratajczak, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, <i>Methods of high efficiency compression for transmission of spatial representation of motion scenes</i> , IEEE International Conference on Multimedia and Expo, ICME 2015, Turyn, Włochy, 29.06-3.07.2015
[Domański'15C]	M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, [FTV AHG] Video and depth multiview test sequences acquired with circular camera arrangement – "Poznan Service" and "Poznan People", ISO/IEC JTC1/SC29/WG11 MPEG2015, M36569, Warszawa, Polska, 20-27.06.2015

[Domański'15D]	M. Domański, K. Klimaszewski, A. Dziembowski, D. Mieloch, A. Łuczak, O. Stankiewicz, K. Wegner, <i>Freeview navigation (FN) anchor generation using 3D-HEVC with depth for "Poznan Blocks" sequence</i> , ISO/IEC JTC1/SC29/WG11 MPEG2015, M36565, Warszawa, Polska, 20-27.06.2015
[Domański'15E]	M. Domański, A. Dziembowski, M. Kurc, A. Łuczak, D. Mieloch, J. Siast, O. Stankiewicz, K. Wegner, <i>Poznan University of Technology test multiview video sequences acquired with circular camera arrangement – "Poznan Team" and "Poznan Blocks" sequences</i> , ISO/IEC JTC1/SC29/WG11 MPEG2015, M35846, Genewa, Szwajcaria, 14-20.02.2015
[Domański'15F]	M. Domański, A. Dziembowski, K. Klimaszewski, A. Łuczak, D. Mieloch, O. Stankiewicz, K. Wegner, <i>Comments on further standarization for free-viewpoint television</i> , ISO/IEC JTC1/SC29/WG11 MPEG2015, M35842, Genewa, Szwajcaria, 14-20.02.2015
[Domański'15G]	M. Domański, K. Klimaszewski, M. Kurc, O. Stankiewicz, K. Wegner, <i>Super-multiview light-field images from Poznan University of Technology</i> , ISO/IEC JTC1/SC29/WG11 MPEG2015, M36566, Warszawa, Polska, 20-27.06.2015
[Domański'16A]	M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, <i>Optimization of camera positions for free-navigation applications</i> , International Conference on Signals and Electronic Systems, ICSES 2016, Kraków, Polska, 5-7.09.2016
[Domański'16B]	M. Domański, M. Bartkowiak, A. Dziembowski, T. Grajek, A. Grzelka, A. Łuczak, D. Mieloch, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, <i>New results in free-viewpoint television systems for horizontal virtual navigation</i> , IEEE International Conference on Multimedia and Expo, ICME 2016, Seattle, Stany Zjednoczone, 11-15.07.2016
[Domański'16C]	M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, <i>Experimental video conding software for free navigation applications</i> , ISO/IEC JTC1/SC29/WG11 MPEG2016, M39527, Chengdu, Chiny, 17-21.10.2016
[Domański'16D]	M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, <i>Coding results for Poznan Fencing 2 and Poznan Blocks 2 test sequences in free navigation scenario</i> , ISO/IEC JTC1/SC29/WG11 MPEG2016, M39215, Chengdu, Chiny, 17-21.10.2016
[Domański'16E]	M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, <i>[FTV AHG]</i> Extended results of Poznan University of Technology proposal for Call for Evidence on free-viewpoint television, ISO/IEC JTC1/SC29/WG11 MPEG2016, M38246, Genewa, Szwajcaria, 30.05-3.06.2016

- [Domański'16F] M. Domański, A. Dziembowski, A. Grzelka, A. Łuczak, D. Mieloch, O. Stankiewicz, K. Wegner, Multiview test video sequences for free navigation exploration obtained using pairs of cameras, ISO/IEC JTC1/SC29/WG11 MPEG2016, M38247, Genewa, Szwajcaria, 30.05-3.06.2016 [Domański'16G] M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, Study on nonuniform distributions of cameras located on an arc, ISO/IEC JTC1/SC29/WG11 MPEG2016, M38248, Genewa, Szwajcaria, 30.05-3.06.2016 [Domański'16H] M. Domański, A. Dziembowski, A. Grzelka, Ł. Kowalski, D. Mieloch, J. Samelak, J. Stankowski, O. Stankiewicz, K. Wegner, *FTV AHG* Technical Description of Poznan University of Technology proposal for Call for Evidence on free-viewpoint television, ISO/IEC JTC1/SC29/WG11 MPEG2016, M37893, San Diego, Stany Zjednoczone, 22-26.02.2016 [Domański'17] M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, K. Klimaszewski, D. Mieloch, R. Ratajczak, O. Stankiewicz, J. Siast, J. Stankowski, K. Wegner, Demonstration of a simple free viewpoint television systems, IEEE International Conference on Image Processing, ICIP 2017, Pekin, Chiny, 17-20.09.2017 [Domański'18A] M. Domański, D. Łosiewicz, T. Grajek, O. Stankiewicz, K. Wegner, A. Dziembowski, D. Mieloch, Extended VSRS for 360 degree video, ISO/IEC JTC1/SC29/WG11 MPEG2018, M41990, Gwangju, Korea Południowa, 22-26.01.2018 [Domański'18B] M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, D. Mieloch, R. Ratajczak, J. Samelak, O. Stankiewicz, J. Stankowski, K. Wegner, Freeviewpoint television demonstration for events, ISO/IEC sports JTC1/SC29/WG11 MPEG2018, M41994, Gwangju, Korea Południowa, 22-26.01.2018 [Domański'18C] M. Domański, A. Dziembowski, T. Grajek, A. Grzelka, D. Mieloch, O. Stankiewicz, J. Stankowski, K. Wegner, Real-time virtual navigation
 - Du Hein²121
- [Du-Hsiu'13] L. Du-Hsiu, H. Hsueh-Ming, L. Yu-Lun, Virtual view synthesis using backward depth warping algorithm, Picture Coding Symposium, PCS 2013, San Jose Stany Zjednoczone, 8-11.12.2013
- [Dubé'18]R. Dubé, M.G. Gollub, H. Sommer, I. Giligschenski, R. Siegwart,
C. Cadena, J. Nieto, Incremental-segment-based localization in 3-D point clouds,
IEEE Robotics and Automation Letters, tom 3, nr 3, 07.2018
- [Dziembowski'14A] A. Dziembowski, Zastosowanie przestrzeni promieni w systemie telewizji swobodnego punktu widzenia, praca magisterska, Politechnika Poznańska, 2014
- **[Dziembowski'14B]** A. Dziembowski, A. Kuehn, A. Łuczak, D. Mieloch, K. Wegner, *Realizacja eksperymentalnego systemu telewizji swobodnego punktu widzenia z łukowym ustawieniem kamer*, Przegląd Telekomunikacyjny, nr 6, s. 161-164, 2014

[Dziembowski'16A]	A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, K. Wegner, M. Domański, <i>Multiview Synthesis – improved view synthesis for virtual navigation</i> , 32nd Picture Coding Symposium, PCS 2016, Norymberga, Niemcy, 4-7.12.2016
[Dziembowski'16B]	A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, M. Domański, Depth map upsampling and refinement for FTV systems, International Conference on Signals and Electronic Systems, ICSES 2016, Kraków, Polska, 5-7.09.2016
[Dziembowski'16C]	A. Dziembowski, M. Domański, A. Grzelka, D. Mieloch, J. Stankowski, K. Wegner, <i>The influence of a lossy compression on the quality of estimated depth maps</i> , 23rd International Conference on Systems, Signals and Image Processing, IWSSIP 2016, Bratysława, Słowacja, 23-25.05.2016
[Dziembowski'16D]	A. Dziembowski, A. Grzelka, D. Mieloch, Wielowidokowa synteza w systemach telewizji swobodnego punktu widzenia, Przegląd Telekomunikacyjny, tom 88, nr 6, s. 233-236, 2016
[Dziembowski'17A]	A. Dziembowski, A. Grzelka, D. Mieloch, Zwiększanie rozdzielczości obrazu i mapy głębi w celu poprany jakości syntezy widoków wirtualnych, Przegląd Telekomunikacyjny, tom 86, nr 6, s. 405-408, 2017
[Dziembowski'17B]	A. Dziembowski, A. Grzelka, D. Mieloch, O. Stankiewicz, M. Domański, <i>Enhancing view synthesis with image and depth map upsampling</i> , International Conference on Systems, Signals and Image Processing, IWSSIP 2017, Poznań, Polska, 22-24.05.2017
[Dziembowski'18A]	A. Dziembowski, J. Stankowski, <i>Szybka synteza widoków wirtualnych w systemach telewizji swobodnego punktu widzenia</i> , Przegląd Telekomunikacyjny, nr 6, s. 330-333, 2018
[Dziembowski'18B]	A. Dziembowski, View and depth preprocessing for view synthesis enhancement, International Journal of Electronics and Telecommunications, tom 64, nr 3, s. 269-275, 2018
[Dziembowski'18C]	A. Dziembowski, M. Domański, <i>Adaptive color correction method in virtual view synthesis</i> , 3DTV Conference 2018, Sztokholm/Helsinki, Szwecja /Finlandia, 3-5.06.2018
[Dziembowski'18D]	A. Dziembowski, J. Stankowski, <i>Real-time CPU-based virtual view synthesis</i> , International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018
[Dziembowski'18E]	A. Dziembowski, J. Samelak, M. Domański, View selection for virtual view synthesis in free navigation systems, International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018
[Dziembowski'18F]	A. Dziembowski, O. Stankiewicz, [MPEG-I Visual] Fast color correction technique for view synthesis, ISO/IEC JTC1/SC29/WG11 MPEG 2018, M43694, Ljubljana, Słowenia, 13-20.07.2018

	Adrian Dziembowski, rozprawa doktorska
[Dziembowski'18G]	A. Dziembowski, D. Mieloch, K. Wegner, O. Stankiewicz, M. Domański, Proposal of enhanced version of View Synthesis Reference Software with multiple input views, ISO/IEC JTC1/SC29/WG11 MPEG 2018, M43694, Ljubljana, Słowenia, 13-20.07.2018
[Fachada'18A]	S. Fachada, D. Bonatto, A. Schenkel, G. Lafruit, <i>Depth image based view synthesis with multiple reference views for virtual reality</i> , 3DTV Conference 2018, Sztokholm/Helsinki, Szwecja /Finlandia, 3-5.06.2018
[Fachada'18B]	S. Fachada, D. Bonatto, A. Schenkel, M. Teratani, G. Lafruit, [MPEG-I Visual] ULB proposal of new test sequences for Windowed-6DoF, ISO/IEC JTC1/SC29/WG11 MPEG 2018, M43694, Ljubljana, Słowenia, 13-20.07.2018
[Fecker'08]	U. Fecker, M. Barkovsky, A. Kaup, <i>Histogram-based prefiltering for luminance and chrominance compensation of multiview video</i> , IEEE Transactions on Circuits and Systems for Video Technology, tom 18, nr 9, s. 1258-1267, 09.2008
[Fehn'03]	C. Fehn, A 3D-TV approach using depth-image-based rendering (DIBR), International Conference on Visualization, Imaging and Image Processing, VIIP 2003, Benalmadena, Hiszpania, 8-10.08.2003
[Fehn'06]	C. Fehn, R. de la Barre, S. Pastoor, <i>Interactive 3-DTV-concepts and key technologies</i> , Proceedings of the IEEE, tom 94, nr 3, s. 524-538, 03.2006
[Feldmann'08]	I. Feldmann, M. Mueller, F. Zilly, R. Tanger, K. Mueller, A. Smolic, P. Kauff, T. Wiegand, <i>HHI test material for 3D video</i> , ISO/IEC JTC1/SC29/WG11 MPEG2017, M15413, Archamps, Francja, 28.04-2.05.2008
[Fezza'14]	S.A. Fezza, M.C. Larabi, K.M. Faraoun, <i>Feature-based color correction of multiview video for coding and rendering enhancement</i> , IEEE Transactions on Circuits and Systems for Video Technology, tom 24, s. 1486-1498, 2014
[Fujii'00]	T. Fujii, T. Kimoto, M. Tanimoto, A new flexible acquisition system of ray-space data for arbitrary objects, IEEE Transactions on Circuits and Systems for Video Technology, tom 10, nr 2, s. 218-224, 03.2000
[Fujii'06]	T. Fujii, K. Mori, K. Takeda, K. Mase, M. Tanimoto, Y. Suenaga, <i>Multipoint measuring system for video and sound – 100-camera and microphone system</i> , IEEE International Conference on Multimedia and Expo, ICME 2006, Toronto, Kanada, 9-12.07.2006
[Gallup'07]	D. Gallup, J.M. Frahm, P. Mordohai, Q. Yang, M. Pollefeys, <i>Real-time plane-sweeping stereo with multiple sweeping directions</i> , IEEE Conference on Computer Vision and Pattern Recognition, CVPR'07, Minneapolis, Stany Zjednoczone, 17-22.07.2007
[Golestani'14]	H.B. Golestani, M. Ghanbari, <i>Window size influence on SSIM fidelity</i> , 7th International Symposium on Telecommunications, IST'2014, Teheran, Iran, 9-11.09.2014

[Goorts'12]	P. Goorts, M. Dumont, S. Rogmans, P. Bekaert, <i>An end-to-end system for free viewpoint video for smooth camera transitions</i> , International Conference on 3D Imaging, IC3D 2012, Liege, Belgia, 3-5.12.2012
[Goorts'14A]	P. Goorts, S. Maesen, M. Dumont, S. Rogmans, P. Bekaert, <i>Free viewpoint video for soccer using histogram-based validity maps in plane sweeping</i> , International Conference on Computer Vision Theory and Applications, VISAPP 2014, Lizbona, Portugalia, 5-8.01.2014
[Goorts'14B]	P. Goorts, <i>The Genk dataset</i> , w: "Real-time adaptive plane sweeping for free viewpoint navigation in soccer scenes", rozprawa doktorska, Hasselt University, Belgia, s. 175-180, 16.06.2014
[Goorts'14C]	P. Goorts, <i>The Barcelona dataset</i> , w: "Real-time adaptive plane sweeping for free viewpoint navigation in soccer scenes", rozprawa doktorska, Hasselt University, Belgia, s. 181-186, 16.06.2014
[Habigt'13]	J. Habigt, K. Diepold, Image completion for view synthesis using Markov random fields and efficient belief propagation, 20th IEEE International Conference on Image Processing, ICIP 2013, Melbourne, Australia, 15-18.08.2013
[Hartley'15]	R. Hartley, A. Zisserman, <i>Multiple view geometry in computer vision</i> , wyd. 2, Cambridge University Press, 2015
[He'12]	K. He, J. Sun, Statistics of patch offsets for image completion, Computer Vision – ECCV 2012, Springer, s. 16-29, 2012
[Held'08]	R.T. Held, M.S. Banks, <i>Misperceptions in stereoscopic displays: a vision science perspective</i> , 5th Symposium on Applied Perception in Graphics and Visualization, APGV'08, Los Angeles, Stany Zjednoczone, 9-10.08.2008
[HEVC]	ISO/IEC, High efficiency coding and media delivery in heterogeneous environments, Part 2: High Efficiency Video Coding, ISO/IEC IS 23008-2, ITU-T Rec. H.265, 2015
[Heyden'05]	A. Heyden, M. Pollefeys, <i>Multiple view geometry</i> , w: G. Medioni, S.B. Kang (ed.) "Emerging Topics in Computer Vision", Prentice Hall, s. 63-75, 2005
[Hirschmüller'07]	H. Hirschmüller, D. Scharstein. <i>Evaluation of cost functions for stereo matching</i> , IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2007, Minneapolis, Stany Zjednoczone, 06.2007
[Ho'08]	Y.S. Ho, E.K. Lee, C. Lee, <i>Multiview video test sequence and camera parameters</i> , ISO/IEC JTC1/SC29/WG11 MPEG2017, M15419, Archamps, Francja, 28.04-2.05.2008
[Hołowko'14]	E. Hołowko, J. Wojsz, R. Sitnik, M. Karaszewski, <i>Color-based algorithm for automatic merging of multiview 3D point clouds</i> , Journal on Computing and Cultural Heritage, JOCCH, tom 7, nr 3, 08.2014

	Adrian Dziembowski, rozprawa doktorska
[Howarth'11]	P.A. Howarth, Potential hazards of viewing 3-D stereoscopic television, cinema and computer games: a review, Ophthalmic & Physiological Optics: The Journal of the British College of Ophthalmic Opticians (Optometrists), tom 31, nr 2, s. 111-122, 03.2011
[Huang'14]	J.B. Huang, S.B. Kang, N. Ahuja, J. Kopf, Image completion using planar structure guidance, ACM Transactions on Graphics – TOG, tom 33, nr 4, 07.2014
[ITU-R BT.500-9]	ITU-R, Methodology for the subjective assessment of the quality of television pictures, Recommendation ITU-R BT.500-9, 11.1998
[ITU-T P.910]	ITU-T, Subjective video quality assessment methods for multimedia applications, Recommendation ITU-T P.910, 04.2008
[Jang'15]	W.D. Jang, T.Y. Chung, J.Y. Sim, C.S. Kim, <i>FDQM: fast quality metric for depth maps without view synthesis</i> , IEEE Transactions on Circuits and Systems for Video Technology, tom 25, nr 7, s. 1099-1112, 2015
[Jin'16]	J. Jin, A. Wang, Y. Zhao, C. Lin, B. Zeng, <i>Region-aware 3D warping for DIBR</i> , IEEE Transactions on Multimedia, tom 18, nr 6, s. 953-966, 2016
[Jorissen'14]	L. Jorissen, P. Goorts, B. Bex, N. Michiels, S. Rogmans, P. Bekaert, G. Lafruit, <i>A qualitative comparison of MPEG view synthesis and light field rendering</i> , 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Budapeszt, Węgry, 2-4.07.2014
[Jun-Te'15]	H. Jun-Te, L. Jin-Jang, Virtual view synthesis for multi-view video plus depth sequences using spatial-temporal information, 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Lizbona, Portugalia, 8-10.07.2015
[Kang'10]	Y.S. Kang, Y.S. Ho, <i>High-quality multi-view depth generation using multiple color and depth cameras</i> , IEEE International Conference on Multimedia and Expo, ICME 2010, Singapur, 19-23.07.2010
[Kawamura'13]	K. Kawamura, H. Sankoh, S. Naito, <i>Requirements and use cases of free-navigation for fly-through experience</i> , ISO/IEC JTC1/SC29/WG11 MPEG2013, M31521, Genewa, Szwajcaria, 28.10-1.11.2013
[Khatiullin'18]	A. Khatiullin, M. Erofeev, D. Vatolin, <i>Fast occlusion filling method for multiview video generation</i> , 3DTV Conference 2018, Sztokholm/Helsinki, Szwecja /Finlandia, 3-5.06.2018
[Kim'13]	C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, M. Gross, Scene reconstruction from high spatio-angular resolution light fields, ACM Transactions on Graphics – TOG, tom 32, nr 4, 07.2013
[Koeppel'16]	M. Koeppel, K. Mueller, T. Wiegand, <i>Filling disocclusions in extrapolated virtual views using hybrid texture synthesis</i> , IEEE Transactions on Broadcasting, tom 62, s. 457-469, 2016

[Komodakis'07]	N. Komodakis, G. Tziritas, Image completion using efficient belief propagation via priority scheduling and dynamic pruning, IEEE Transactions on Image Processing, tom 16, nr 11, s. 2649-2661, 2007
[Koppal'14]	S.J. Koppal, Lambertian reflectance, w: K. Ikeuchi (ed.) "Encyclopedia of Computer Vision", Springer, s. 441-443, 2014
[Kovacs'15A]	P.T. Kovacs, A. Fekete, K. Lackner, V.K. Adhikarla, A. Zare, T. Balogh, [FTV AHG] Big Buck Bunny light-field test sequences, ISO/IEC JTC1/SC29/WG11 MPEG2013, M35721, Genewa, Szwajcaria, 16- 20.02.2015
[Kovacs'15B]	P.T. Kovacs, A. Fekete, K. Lackner, V.K. Adhikarla, A. Zare, T. Balogh, [FTV AHG] Big Buck Bunny light-field test sequences, ISO/IEC JTC1/SC29/WG11 MPEG2013, M36500, Warszawa, Polska, 22- 26.06.2015
[Kumar'05]	S. Kumar, M. Biswas, S.J. Belongie, T.Q. Nguyen, <i>Spatio-temporal texture</i> synthesis and image inpainting for video applications, IEEE International Conference on Image Processing, ICIP 2005, Genua, Włochy, 11-14.08.2005
[Lafruit'15A]	G. Lafruit, K. Wegner, M. Tanimoto, <i>FTV software framework</i> , ISO/IEC JTC1/SC29/WG11 MPEG2015, N15349, Warszawa, Polska, 20-27.06.2015
[Lafruit'15B]	G. Lafruit, K. Wegner, M. Tanimoto, <i>Draft Call for Evidence on FTV</i> , ISO/IEC JTC1/SC29/WG11 MPEG2013, N15095, Genewa, Szwajcaria, 16-20.02.2015
[Lafruit'16]	G. Lafruit, M. Domański, K. Wegner, T. Grajek, T. Senoh, J. Jung, P. Kovacs, P. Goorts, L. Jorissen, A. Munteanu, B. Ceulemans, P. Carballeira, S. Garcia, M. Tanimoto, <i>New visual coding exploration in</i> <i>MPEG: super-multiview and free navigation in free viewpoint TV</i> , IST Electronic Imaging, Stereoscopic Displays and Applications XXVII, San Francisco, Stany Zjednoczone, 14-18.02.2016
[Lai'17]	Y. Lai, X. Tang, X. Lou, <i>Three-dimensional video inpainting using gradient fusion and clustering</i> , International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery, ICNC-FSKD 2017, Guilin, Chiny, 29-31.07.2017
[Lee'08]	C. Lee, Y.S. Ho, Boundary filtering on synthesized views of 3D video, 2nd International Conference on Future Generation Communication and Networking, FGCN 2008, Sanya, Chiny, 13-15.12.2008
[Lee'13A]	B. Lee, <i>Three-dimensional displays, past and present</i> , Physics today, tom 66, nr 4, s. 36-41, 2013
[Lee'13B]	J.S. Lee, L. Goldmann, T. Ebrahimi, <i>Paired comparison-based subjective quality</i> assessment of stereoscopic images, Multimedia Tools and Applications, tom 67, nr 1, s. 31-48, 11.2013

	Adrian Dziembowski, rozprawa doktorska
[Lee'15]	C.C. Lee, A. Tabatabai, K. Tashiro, <i>Free viewpoint video (FVV) survey and future research direction</i> , APSIPA Transactions on Signal and Information Processing, tom 4, 10.2015
[LeMeur'13]	O. LeMeur, M. Ebdelli, C. Guillemot, <i>Hierarchical super-resolution-based inpainting</i> , IEEE Transactions on Image Processing, tom 22, nr 10, s. 3779-3790, 10.2013
[Levin'03]	A. Levin, A. Zomet, Y. Weiss, <i>Learning how to inpaint from global image statistics</i> , 9th International Conference on Computer Vision, Nicea, Francja, 13-16.10.2003
[Levoy'06]	M. Levoy, Light fields and computational imaging, IEEE Computer, tom 39, nr 8, s. 46-55, 08.2006
[Li'18]	S. Li, C. Zhu, M.T. Sun, <i>Hole filling with multiple reference views in DIBR view</i> synthesis, IEEE Transactions on Multimedia, tom 20, nr 8, s. 1948-1959, 08.2018
[Lipski'14]	C. Lipski, F. Klose, M. Magnor, <i>Correspondence and depth-image based rendering a hybrid approach for free-viewpoint video</i> , IEEE Transactions on Circuits and Systems for Video Technology, tom 24, nr 6, s. 942-951, 06.2014
[Liu'12]	H. Liu, X. Li, K. Chen, <i>Global-background based view synthesis approach for multi-</i> view video, 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Zurych, Szwajcaria, 15- 17.10.2012
[Lowe'04]	D.G. Lowe, <i>Distinctive image features from scale-invatiant keypoints</i> , International Journal of Computer Vision, tom 60, nr 2, s. 91-110, 11.2004
[Lu'13]	S. Lu, J. Hanca, A. Munteanu, P. Schelkens, <i>Depth-based view synthesis using pixel-level image inpainting</i> , 18th International Conference on Digital Signal Processing, DSP 2013, Fira, Grecja, 1-3.07.2013
[Lu'15]	S. Lu, B. Ceulemans, A. Munteanu, P. Schelkens, <i>Spatio-temporally consistent color and structure optimization for multiview video color correction</i> , IEEE Transactions on Multimedia, tom 17, nr 5, s. 577-590, 2015
[Luo'17]	G. Luo, Y. Zhu, Foreground removal approach for hole filling in 3D video and FVV synthesis, IEEE Transactions on Circuits and Systems for Video Technology, tom 27, nr 10, s. 2118-2131, 10.2017
[Malpica'09]	W. Malpica, A.C. Bovik, <i>SSIM based range image quality assessment,</i> 4th International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, Stany Zjednoczone, 15-16.01.2009
[Mantiuk'12]	R.K. Mantiuk, A. Tomaszewska, R. Mantiuk, <i>Comparison of four subjective methods for image quality assessment</i> , Computer Graphics Forum, tom 31, nr 8, s. 2478-2491, 12.2012

[Mao'13]	Y. Mao, G. Cheung, A. Ortega, Y. Ji, <i>Expansion hole filling in depth-image-based rendering using graph-based interpolation</i> , IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, Kanada, 26-31.05.2013
[Mao'14]	Y. Mao, G. Cheung, Y. Ji, Image interpolation for DIBR view synthesis using graph Fourier transform, 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Budapeszt, Węgry, 2-4.07.2014
[Merkle'07]	P. Merkle, A. Smolic, K. Mueller, T. Wiegand, <i>Multi-view Video plus Detph representation and coding</i> , IEEE International Conference on Image Processing, ICIP 2007, San Antonio, Stany Zjednoczone, 16-19.10.2007
[Middlebury]	http://vision.middlebury.edu/stereo - strona dostępna 1.08.2018
[Mieloch'16]	D. Mieloch, A. Dziembowski, A. Grzelka, Segmentacja obrazu w estymacji map głębi, Przegląd Telekomunikacyjny, tom 88, nr 6, s. 241-244, 2016
[Mieloch'17A]	D. Mieloch, A. Dziembowski, A. Grzelka, O. Stankiewicz, M. Domański, <i>Graph-based multiview depth estimation using segmentation</i> , IEEE International Conference on Multimedia and Expo, ICME 2017, Hongkong, Chiny, 10-14.07.2017
[Mieloch'17B]	D. Mieloch, A. Dziembowski, A. Grzelka, O. Stankiewicz, M. Domański, <i>Temporal enhancement of graph-based depth estimation method</i> , International Conference on Systems, Signals and Image Processing, IWSSIP 2017, Poznań, Polska, 22-24.05.2017
[Mieloch'17C]	D. Mieloch, A. Dziembowski, A. Grzelka, <i>Estymacja głębi dla systemów wielowidokowych</i> , Przegląd Telekomunikacyjny, tom 86, nr 6, s. 479-482, 2017
[Miller'06]	G. Miller, J. Starck, A. Hilton, <i>Projective surface refinement for free-viewpoint video</i> , 3rd European Conference on Visual Media Production, CVMP 2006, Londyn, Wielka Brytania, 29-30.11.2006
[Mori'09]	Y. Mori, N. Fukushima, T. Yendo, T. Fujii, M. Tanimoto, <i>View generation with 3D warping using depth information for FTV</i> , Signal Processing: Image Communication, tom 24, nr. 1-2, s. 65-72, 01.2009
[Mueller'08]	K. Mueller, A. Smolic, K. Dix, P. Merkle, P. Kauff, T. Wiegand, <i>View</i> synthesis for advanced 3D video systems, EURASIP Journal on Image and Video Processing, tom 2008, s. 1-11, 2008
[Mueller'11]	K. Mueller, P. Merkle, T. Wiegand, <i>3D video representation using depth maps</i> , Proceedings of the IEEE, tom 99, nr 4, s. 643-656, 04.2011
[Mueller'13]	K. Mueller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. Hunn Rhee, G. Tech, M. Winken, T. Wiegand, <i>3D high-efficiency video coding for multi-view video and depth data</i> , IEEE Transactions on Image Processing, tom 22, s. 3366-3378, 2013

	Adrian Dziembowski, rozprawa doktorska		
[Ndjiki-Nya'11]	P. Ndjiki-Nya, M. Koeppel, D. Doshkov, H. Lakshman, P. Merkle, K. Mueller, T. Wiegand, <i>Depth image-based rendering with advanced texture synthesis for 3D video</i> , IEEE Transactions on Multimedia, tom 13, nr 3, s. 453-465, 06.2011		
[Newcombe'11]	R.A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A.J. Davison, P. Kohi, J. Shotton, S. Hodges, A. Fitzgibbon, <i>KinectFusion: real-time dense surface mapping and tracking</i> , 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2011, 26-29.10.2011		
[Oh'09]	K.J. Oh, S. Yea, Y.S. Ho, Hole filling method using depth based inpainting for view synthesis in free viewpoint television and 3-D video, Picture Coding Symposium, PCS 2009, Chicago, Stany Zjednoczone, 6-8.05.2009		
[Palacio'17]	K. Palacio, M. Mejia-Pesantez, L. Munoz-Guillen, M. Espinoza-Mejia, <i>Current challenges of interactive digital television</i> , IEEE Second Ecuador Technical Chapters Meeting, ETCM 2017, Salinas, Ekwador, 16-20.10.2017		
[Pearl'82]	J. Pearl, Reverend Bayes on inference engines: A distributed hierarchical approach, Second National Conference on Artificial Intelligence, AAAI-82, Pittsburgh, Stany Zjednoczone, 1982		
[Plath'13]	N. Plath, S. Knorr, L. Goldmann, T. Sikora, <i>Adaptive image warping for hole prevention in 3D view synthesis</i> , IEEE Transactions on Image Processing, tom 22, nr 9, s. 3420-3432, 2013		
[Purica'15]	A.I. Purica, M. Cagnazzo, B. Pesquet-Popescu, F. Dufaux, B. Ionescu, <i>A distortion evaluation framework in 3D video view synthesis</i> , International Conference on 3D Imaging, IC3D, Liège, Belgia, 14-15.12.2015		
[Reinhard'01]	E. Reinhard, M. Adhikhmin, B. Gooch, P. Shirley, <i>Color transfer between images</i> , IEEE Computer Graphics and Applications, tom 21, nr 5, s. 31-41, 09-10.2001		
[Rogmans'09]	S. Rogmans, J. Lu, P. Bekaert, G. Lafruit, Real-time stereo-based view synthesis algorithms: a unified framework and evaluation on commodity GPUs, Signal Processing: Image Communication, tom 24, nr. 1-2, s. 49-64, 01.2009		
[Scharstein'02]	D. Scharstein, R. Szeliski, <i>A taxonomy and evaluation of dense two-frame stereo correspondence algorithms</i> , International Journal of Computer Vision, tom 47, nr. 1-3, s. 7-42, 04.2002		
[Scharstein'03]	D. Scharstein, R. Szeliski. <i>High-accuracy stereo depth maps using structured light</i> , IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2003, Madison, Stany Zjednoczone, 18-20.06.2003		
[Scharstein'14]	D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, P. Westling. <i>High-resolution stereo datasets with subpixel-accurate ground truth</i> , German Conference on Pattern Recognition, GCPR 2014, Münster, Niemcy, 2-5.09.2014		

[Schmeing'15]	M. Schmeing, X. Jiang, <i>Faithful disocclusion filling in deph image based rendering using superpixel-based inpainting</i> , IEEE Transactions on Multimedia, tom 17, nr 12, s. 2160-2173, 12.2015
[Seitz'06]	S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, R. Szeliski, <i>A comparison and evaluation of multi-view stereo reconstruction algorithms</i> , IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'06, Nowy Jork, Stany Zjednoczone, 17-22.06.2006
[Senoh'16]	T. Senoh, A. Ishikawa, M. Okui, K. Yamamoto, N. Inoue, <i>FTV AHG:</i> fast DERS (depth estimation software), ISO/IEC JTC1/SC29/WG11 MPEG2013, M37951, San Diego, Stany Zjednoczone, 22-26.02.2016
[Senoh'17A]	T. Senoh, K. Yamamoto, N. Tetsutani, H. Yasuda, K. Wegner, View Synthesis Reference Software (VSRS) 4.2 with improved inpainting and hole filling, ISO/IEC JTC1/SC29/WG11 MPEG2013, M40657, Hobart, Australia, 3-7.04.2017
[Senoh'17B]	T. Senoh, K. Yamamoto, N. Tetsutani, H. Yasuda, MPEG-I Visual: EE results on light-field compression, ISO/IEC JTC1/SC29/WG11 MPEG2017, M41784, Makau, Chiny, 23-27.10.2017
[Shao'08]	F. Shao, G. Jiang, M. Yu, New color correction method of multi-view images for view rendering in free-viewpoint television, WSEAS Transactions on Computers, tom 7, s. 569-578, 2008
[Shao'12]	F. Shao, G. Jiang, M. Yu, K. Chen, Y.S. Ho, <i>Asymmetric coding of multi-view video plus depth based 3-d video for view rendering</i> , IEEE Transactions on Multimedia, tom 14, nr 1, s. 157-167, 02.2012
[Skupin'14]	R. Skupin, T. Borgmann, T. Sikora, <i>Multiview point cloud filtering for spatiotemporal consistency</i> , International Conference on Computer Vision Theory and Applications, VISAPP 2014, Lizbona, Portugalia, 5-8.01.2014
[Smolic'05]	A. Smolic, K. Mueller, P. Merkle, M. Kautzner, T. Wiegand, <i>3D video objects for interactive applications</i> , 13th European Signal Processing Conference, EUSIPCO 2005, Antalya, Turcja, 4-8.08.2005
[Solh'12]	M. Solh, G. AlRegib, <i>Hierarchical hole-filling for depth-based view synthesis in FTV and 3D video</i> , IEEE Journal of Selected Topics in Signal Processing, tom 6, nr 5, s. 495-504, 08.2012
[Stankiewicz'18]	O. Stankiewicz, M. Domański, A. Dziembowski, A. Grzelka, D. Mieloch, J. Samelak, <i>A free-viewpoint television system for horizontal virtual navigation</i> , IEEE Transactions on Multimedia, tom 20, nr 8, s. 2182-2195, 08.2018
[Stankowski'10]	J. Stankowski, K. Klimaszewski, O. Stankiewicz, K. Wegner, M. Domański, <i>Preprocessing methods used for Poznan 3D/FTV test sequences</i> , ISO/IEC JTC1/SC29/WG11 MPEG2010, M17174, Kioto, Japonia, 18-22.01.2010

Adrian Dziembowski, rozprawa doktorska	
[Stankowski'15]	J. Stankowski, Ł. Kowalski, J. Samelak, M. Domański, T. Grajek, K. Wegner, <i>3D-HEVC extension for circular camera arrangements</i> , 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Lizbona, Portugalia, 8-10.07.2015
[Sullivan'13]	G.J. Sullivan, J.M. Boyce, Y. Chen, J.R. Ohm, C.A. Segall, A. Vetro, <i>Standarized extensions of High Efficiency Video Coding (HEVC)</i> , IEEE Journal of Selected Topics in Signal Processing, tom 7, nr 6, s. 1001-1016, 12.2013
[Tagare'91]	H.D. Tagare, R.J.P. deFigueiredo, <i>A theory of photometric stereo for a class of diffuse non-Lambertian Surfaces</i> , IEEE Transactions on Pattern Analysis and Machine Intelligence, tom 13, nr 2, s. 133-152, 02.1991
[Tanimoto'02]	M. Tanimoto, T. Fujii, <i>FTV – Free Viewpoint Television</i> , ISO/IEC JTC1/SC29/WG11 MPEG2002, M8595, Klagenfurt, Austria, 22-26.07.2002
[Tanimoto'05]	M. Tanimoto, <i>FTV (free viewpoint television) creating ray-based image engineering</i> , IEEE International Conference on Image Processing, ICIP 2005, Genua, Włochy, 11-14.08.2005
[Tanimoto'06]	M. Tanimoto, Overview of free viewpoint television, Signal Processing: Image Communication, tom 21, nr 6, s. 454-461, 06.2006
[Tanimoto'08A]	M. Tanimoto, T. Fujii, K. Suzuki, Reference software of depth estimation and view synthesis for FTV/3DV, ISO/IEC JTC1/SC29/WG11 MPEG2008, M15836, Busan, Korea Południowa, 13-17.10.2008
[Tanimoto'08B]	M. Tanimoto, T. Fujii, N. Fukushima, 1D parallel test sequences for MPEG- FTV, ISO/IEC JTC1/SC29/WG11 MPEG2017, M15378, Archamps, Francja, 28.04-2.05.2008
[Tanimoto'09A]	M. Tanimoto, T. Fujii, K. Suzuki, View synthesis Algorithm in View Synthesis Reference Software 2.0 (VSRS2.0), ISO/IEC JTC1/SC29/WG11 MPEG2009, M16090, Lozanna, Szwajcaria, 26-28.01.2009
[Tanimoto'09B]	M. Tanimoto, <i>Overview of FTV (free-viewpoint television)</i> , IEEE International Conference on Multimedia and Expo, ICME 2009, Nowy Jork, Stany Zjednoczone, 28.06-3.07.2009
[Tanimoto'10]	M. Tanimoto, FTV (Free-Viewpoint TV), IEEE International Conference on Image Processing, ICIP 2010, Hongkong, Chiny, 26-29.08.2010
[Tanimoto'12]	M. Tanimoto, M.P. Tehrani, T. Fujii, T. Yendo, FTV for 3-D spatial communication, Proceedings of the IEEE, tom 100, nr 4, s. 905-917, 04.2012
[Tanimoto'13]	M. Tanimoto, T. Senoh, S. Naito, S. Shimizu, H. Horimai, M. Domański, A. Vetro, M. Preda, K. Mueller, <i>Proposal on a new activity for the third phase of FTV</i> , ISO/IEC JTC1/SC29/WG11 MPEG2013, M30232, Wiedeń, Austria, 29.07-2.08.2013

[Tehrani'18]	M. Tehrani, S. Mikawa, S. Fujita, T. Fujii, [MPEG-I Visual] Introduction to a new test sequence "Tunnel_Train_2" captured by light field video camera, ISO/IEC JTC1/SC29/WG11 MPEG2018, M41995, Gwangju, Korea Południowa, 22-26.01.2018
[Telea'04]	A. Telea, <i>An image inpainting technique based on the fast marching method</i> , Journal of Graphics, GPU and Game Tools, tom 9, nr 1, s. 23-34, 2004
[Teratani'18]	M. Teratani, T. Senoh, B. Kroon, K. Wegner, B. Salahieh, D. Didier, J. Jung, G. Bang, Y. Sun, M. Tanimoto, G. Lafruit, L. Yu, <i>Overview of MPEG-I Visual test materials</i> , ISO/IEC JTC1/SC29/WG11 MPEG2018, N17718, Ljubljana, Słowenia, 13-20.07.2018.
[Tezuka'15]	T. Tezuka, M. Tehrani, K. Suzuki, K. Takahashi, T. Fujii, View synthesis using superpixel based inpainting capable of occlusion handling and hole filling, Picture Coding Symposium, PCS 2015, Cairns, Australia, 31.05-3.06.2015
[Uemori'08]	T. Uemori, T. Yendo, T. Fujii, M. Tanimoto, <i>View generation by ray-space method in circular camera setup for FTV</i> , 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Stambuł, Turcja, 28-30.05.2008
[Um'08]	G.M.Um, G. Bang, N. Hur, J. Kim, Y.S. Ho, <i>3D video test material for outdoor scene</i> , ISO/IEC JTC1/SC29/WG11 MPEG2017, M15371, Archamps, Francja, 28.04-2.05.2008
[Um'11]	G.M. Um, T. Kim, G. Bang, N. Hur, E.K. Lee, J.I. Jung, Y.S. Kang, G.Y. Lee, Y.S. Ho, <i>Multi-view 3D video acquisition using hybrid cameras with beam splitter</i> , 3DTV-Conference: The True Vision – Capture, Transmission and Display of 3D Video (3DTV-CON), Antalya, Turcja, 16-18.05.2011
[Urey'11]	H. Urey, K.V. Chellappan, E. Erden, P. Surman, <i>State of the art in stereoscopic and autostereoscopic displays</i> , Proceedings of the IEEE, tom 99, nr 4, s. 540-555, 2011
[vanDijk'97]	A.M. van Dijk, J.B. Martens, Subjective quality assessment of compressed images, Signal Processing, tom 58, nr 3, s. 235-252, 05.1997
[Vetro'11]	A. Vetro, T. Wiegand, G.J. Sullivan, Overview of the stereo and Multiview Video Coding extensions of the H.264/MPEG-4 AVC standard, Proceedings of the IEEE, tom 99, nr 4, s. 626-642, 2011
[Wang'04]	Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, <i>Image quality assessment:</i> from error visibility to structural similarity, IEEE Transactions on Image Processing, tom 13, nr 4, s. 600-612, 04.2004
[Wang'11]	O. Wang, M. Lang, M. Frei, A.Hornung, A. Smolic, M. Gross, <i>StereoBrush:</i> <i>interactive 2D to 3D conversion using discontinuous warps</i> , 8th Eurographics Symposium on Sketch-Based Interfaces and Modelling, Vancouver, Kanada, 5-7.08.2011
[Wang'12]	J. Wang, L.A. Roeningen, Real time believable stereo and virtual view synthesis engine for autostereoscopic display, International Conference on 3D Imaging, IC3D 2012, Liege, Belgia, 3-5.12.2012
	Adrian Dziembowski, rozprawa doktorska
----------------	---
[Warmiński'04]	K. Warmiński, Z. Pióro, R. Sitnik, <i>Concept of interactive television (iTVP)</i> platform, International Workshop on Systems, Signals and Image Processing, IWSSIP'04, Poznań, Polska, 13-15.09.2004
[Wegner'14]	K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, First version of depth maps for Poznan Blocks multiview video test sequence, ISO/IEC JTC1/SC29/WG11 MPEG2014, M32248, San Jose, Stany Zjednoczone, 13-17.01.2014
[Wegner'17A]	K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, <i>Exploration experiments on omnidirectional 6-DoF/3-DoF+ rendering</i> , ISO/IEC JTC1/SC29/WG11 MPEG2017, M41807, Makau, Chiny, 23-27.10.2017
[Wegner'17B]	K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, Evaluation of step-in/step-out capability of state-of-the-art view synthesis technology, ISO/IEC JTC1/SC29/WG11 MPEG2017, M40809, Turyn, Włochy, 17- 21.07.2017
[Wegner'17C]	K. Wegner, O. Stankiewicz, A. Dziembowski, D. Mieloch, M. Domański, Omnidirectional 6-DoF/3-DoF+ rendering, ISO/IEC JTC1/SC29/WG11 MPEG2017, M40806, Turyn, Włochy, 17-21.07.2017
[Wegner'18]	K. Wegner, D. Łosiewicz, T. Grajek, O. Stankiewicz, A. Dziembowski, M. Domański, <i>Omnidirectional view synthesis and test images</i> , International Conference on Signals and Electronic Systems, ICSES 2018, Kraków, Polska, 10-12.09.2018
[Wei'13]	K.C. Wei, Y.L. Huang, S.Y. Chien, <i>Point-based model construction for free-viewpoint TV</i> , IEEE Third International Conference on Consumer Electronics, ICCE 2013, Berlin, Niemcy, 8-11.08.2013
[Wu'17]	P. Wu, Y. Liu, M. Ye, J. Li, S. Du, <i>Fast and adaptive 3d reconstruction with extensively high completeness</i> , IEEE Transactions on Multimedia, tom 19, nr 2, s. 266-278, 02.2017
[Xu'10]	W. Xu, J. Mulligan, Performance evaluation of color correction approaches for automatic multi-view image and video stitching, IEEE Conference on Computer Vision and Pattern Recognition, CVPR'10, San Francisco, Stany Zjednoczone, 13-18.06.2010
[Yamamoto'07]	K. Yamamoto, M. Kitahira, H. Kimata, T. Yendo, <i>Multiview video coding using view interpolation and color correction</i> , IEEE Transactions on Circuits and Systems for Video Technology, tom 17, s. 1436-1449, 2007
[Yang'10]	L. Yang, T. Yendo, M.P. Tehrani, T. Fujii, M. Tanimoto, Artifact reduction using reliability reasoning for image generation of FTV, Journal of Visual Communication and Image Representation, tom 21, nr. 5-6, s. 542-560, 2010
[Yao'13]	Q. Yao, K. Takahashi, T. Fujii, Overcomplete compressed sensing of ray-space for generating free viewpoint images, Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2013, Kaohsiung, Tajwan, 29.10-1.11.2013

Bibliografia

[Yao'14]	C. Yao, T. Tillo, Y. Zhao, J. Xiao, H. Bai, C. Lin, <i>Depth map driven hole filling algorithm exploiting temporal correlation information</i> , IEEE Transactions on Broadcasting, tom 60, nr 2, s. 510-522, 06.2014
[Yao'16]	L. Yao, Y. Liu, W. Xu, Real-time virtual view synthesis using light field, EURASIP Journal on Image and Video Processing, tom 2016, s. 1-10, 2016
[Ye'17]	S. Ye, S.P. Lu, A. Munteanu, <i>Color correction for large-baseline multiview video</i> , Signal Processing: Image Communication, tom 53, s. 40-50, 04.2017
[Zhao'11]	Y. Zhao, C. Zhu, Z. Chen, D. Tian, L. Yu, Boundary artifact reduction in view synthesis of 3D video: from perspective of texture-depth alignment, IEEE Transactions on Broadcasting, tom 57, nr 2, s. 510-522, 06.2011
[Zhou'13]	P. Zhou, L. Yu, G. Zhong, <i>The non-Lambertian reflection in plenoptic sampling</i> , 20 th IEEE International Conference on Image Processing, ICIP 2013, Melbourne, Australia, 15-18.09.2013
[Zhu'13]	C. Zhu, S. Li, <i>A new perspective on hole generation and filling in DIBR based view synthesis</i> , Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP, Pekin, Chiny, 16-18.10.2013
[Zhu'16A]	C. Zhu, S. Li, Depth image based view synthesis: new insights and perspectives on hole generation and filling, IEEE Transactions on Broadcasting, tom 62, nr 1, s. 82-93, 03.2016
[Zhu'16B]	C. Zhu, S. Li, <i>Multiple reference views for hole reduction in DIBR view synthesis</i> , IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, BMSB, Pekin, Chiny, 25-27.06.2014
[Zinger'10]	S. Zinger, Q. Do, P. de With, <i>Free-viewpoint depth image based rendering</i> , Journal of Visual Communication and Image Representation, tom 21, nr. 5-6, s. 533-541, 2010
[Zitnick'04]	C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, <i>High-quality video view interpolation using a layered representation</i> , ACM Transactions on Graphics, tom 3, nr 23, s. 600-608, 08.2004
[Zou'14]	F. Zou, D. Tian, A. Vetro, H. Sun, O.C. Au, S. Shimizu, <i>View synthesis prediction in the 3D video coding extensions of AVC and HEVC</i> , IEEE Transactions on Circuits and Systems for Video Technology, tom 24, nr 10, s. 1696-1708, 2014

ANEKS: SZCZEGÓŁOWE WYNIKI EKSPERYMENTALNE

A1. ZALETY SYNTEZY Z NAJBLIŻSZYCH KAMER

Patrz: rozdział 4.2.3.

BBB BUTTERFLY

Tabela A1. Jakość widoku wirtualnego 45 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

				1. widok r	zeczywisty		
		06	19	32	58	71	84
	84	21,69	23,13	25,89	26,46	20,58	16,86
ty .	71	23,68	24,84	26,83	26,46	20,58	20,58
lobi	58	26,99	27,54	29,10	26,46	26,46	26,46
czy	32	26,40	26,40	26,40	29,10	26,83	25,89
	19	24,12	24,12	26,40	27,54	24,84	23,13
	06	23,74	24,12	26,40	26,99	23,68	21,69

Tabela A2. Jakość widoku wirtualnego 45 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

				1. widok r	zeczywisty		
		06	19	32	58	71	84
	84	21,01	22,02	23,84	22,15	19,28	15,94
ty .	71	23,66	24,01	25,38	24,80	20,69	19,28
dok wist	58	29,18	29,52	31,95	28,25	24,80	22,15
wi	32	26,96	27,74	27,25	31,95	25,38	23,84
rze	19	23,85	22,98	27,74	29,52	24,01	22,02
	06	19,90	23,85	26,96	29,18	23,66	21,01

BBB FLOWERS

Tabela A3. Jakość widoku wirtualnego 45 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

				1. widok r	zeczywisty		
		06	19	32	58	71	84
	84	18,37	20,28	23,01	21,06	18,40	17,26
ty .	, 71	19,76	21,44	23,53	21,06	18,40	18,40
idok wist	58	21,86	22,86	25,08	21,06	21,06	21,06
. wi	32	22,83	22,83	22,83	25,08	23,53	23,01
LZ 2	19	18,64	18,64	22,83	22,86	21,44	20,28
	06	16,23	18,64	22,83	21,86	19,76	18,37

Tabela A4. Jakość widoku wirtualnego 45 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

		1. widok rzeczywisty									
		06	19	32	58	71	84				
	84	17,42	19,69	22,01	19,88	17,47	15,51				
ty	71	18,69	20,38	22,16	19,47	17,04	17,47				
idok wis	58	20,54	21,46	22,68	18,98	19,47	19,88				
czy	32	20,70	20,87	19,96	22,68	22,16	22,01				
rz.	19	18,37	18,18	20,87	21,46	20,38	19,69				
	06	16,11	18,37	20,70	20,54	18,69	17,42				

POZNAN_BLOCKS

Tabela A5. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	18,08	18,40	18,64	21,03	24,03	21,85	20,22	18,47	16,75
Ŷ	8	18,95	19,14	19,33	21,79	24,06	21,85	20,22	18,45	18,47
wist	7	20,11	20,37	20,58	23,42	24,06	21,85	20,20	20,22	20,22
czy	6	21,19	21,32	21,58	23,98	24,06	21,85	21,85	21,85	21,85
rze	5	22,92	23,03	23,29	25,20	23,95	24,06	24,06	24,06	24,03
dok	3	24,03	23,93	23,93	23,93	25,20	23,98	23,42	21,79	21,03
wi	2	19,90	19,79	19,79	23,93	23,29	21,58	20,58	19,33	18,64
5	1	19,62	19,53	19,79	23,93	23,03	21,32	20,37	19,14	18,40
	0	18,92	19,62	19,90	24,03	22,92	21,19	20,11	18,95	18,08

Tabela A6. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	16,25	17,31	17,84	19,55	22,15	19,56	17,73	16,79	16,09
N	8	16,56	17,78	18,38	20,26	22,38	19,60	17,96	16,84	16,79
wist	7	17,65	19,24	20,17	23,07	22,54	19,94	18,57	17,96	17,73
czy	6	18,61	20,22	21,28	23,80	22,75	20,29	19,94	19,60	19,56
rze	5	20,89	22,48	23,36	25,55	22,62	22,75	22,54	22,38	22,15
dok	3	21,61	21,82	21,35	21,69	25,55	23,80	23,07	20,26	19,55
wi	2	18,10	18,33	18,08	21,35	23,36	21,28	20,17	18,38	17,84
5	1	16,88	17,08	18,33	21,82	22,48	20,22	19,24	17,78	17,31
	0	16,36	16,88	18,10	21,61	20,89	18,61	17,65	16,56	16,25

POZNAN_BLOCKS2

Tabela A7. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	26,36	26,61	28,02	28,21	29,26	27,34	26,65	24,88	25,04
Ń	8	25,98	26,07	28,05	28,23	27,11	26,47	25,61	24,39	24,88
wist	7	26,54	26,63	28,54	28,72	27,11	25,55	25,22	25,61	26,65
czy	6	27,20	26,95	29,34	29,57	27,11	25,55	25,55	26,47	27,34
rze	5	28,26	27,77	29,81	29,87	27,11	27,11	27,11	27,11	29,26
dok	3	29,54	29,54	29,54	29,54	29,87	29,57	28,72	28,23	28,21
wie	2	28,97	28,97	28,97	29,54	29,81	29,34	28,54	28,05	28,02
5.	1	25,76	25,76	28,97	29,54	27,77	26,95	26,63	26,07	26,61
	0	25,43	25,76	28,97	29,54	28,26	27,20	26,54	25,98	26,36

Tabela A8. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	25,91	25,94	27,30	27,54	27,56	23,93	24,46	22,78	22,41
Ń	8	25,36	25,26	26,68	27,16	27,23	24,43	24,22	22,29	22,78
wist	7	25,80	25,79	27,04	27,29	27,08	24,54	24,05	24,22	24,46
rzeczyv	6	26,14	26,08	27,73	27,87	26,95	24,32	24,54	24,43	23,93
	5	27,39	27,43	27,99	28,41	26,78	26,95	27,08	27,23	27,56
dok	3	21,00	20,38	20,21	20,80	28,41	27,87	27,29	27,16	27,54
wie	2	20,40	19,89	20,00	20,21	27,99	27,73	27,04	26,68	27,30
6	1	20,01	19,62	19,89	20,38	27,43	26,08	25,79	25,26	25,94
	0	19,94	20,01	20,40	21,00	27,39	26,14	25,80	25,36	25,91

POZNAN_FENCING2

Tabela A9. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	25,34	25,48	26,98	26,78	27,04	24,99	25,63	23,72	24,00
Ŕ	8	25,10	25,04	26,87	26,65	25,50	24,23	24,74	23,11	23,72
wist	7	26,13	25,93	27,63	27,32	24,65	23,32	23,71	24,74	25,63
czy	6	25,59	25,46	27,05	26,87	24,65	23,00	23,32	24,23	24,99
rzec	5	27,15	26,80	27,75	27,54	24,65	24,65	24,65	25,50	27,04
lok	3	26,72	26,72	26,72	26,72	27,54	26,87	27,32	26,65	26,78
wio	2	26,88	26,88	26,88	26,72	27,75	27,05	27,63	26,87	26,98
5.	1	24,11	24,11	26,88	26,72	26,80	25,46	25,93	25,04	25,48
	0	24,21	24,11	26,88	26,72	27,15	25,59	26,13	25,10	25,34

Tabela A10. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	24,68	24,77	27,17	26,80	24,94	22,93	23,18	21,75	21,80
Ŕ	8	24,10	24,11	26,48	26,18	24,49	22,47	22,72	21,14	21,75
wist	7	24,25	24,11	25,74	25,54	24,24	22,55	22,41	22,72	23,18
czy	6	23,74	23,59	25,31	25,14	23,82	21,78	22,55	22,47	22,93
rze	5	25,14	25,00	26,15	26,02	22,69	23,82	24,24	24,49	24,94
dok	3	26,25	26,51	26,53	26,01	26,02	25,14	25,54	26,18	26,80
wi	2	26,31	26,37	26,34	26,53	26,15	25,31	25,74	26,48	27,17
5	1	22,56	22,12	26,37	26,51	25,00	23,59	24,11	24,11	24,77
	0	22,13	22,56	26,31	26,25	25,14	23,74	24,25	24,10	24,68

POZNAN_SERVICE2

Tabela A11. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	20,60	21,14	23,44	23,75	26,73	22,22	21,96	20,08	20,54
czywisty	8	20,38	20,90	23,16	23,51	26,40	22,02	21,77	19,97	20,08
	7	21,15	21,58	23,81	24,15	25,24	21,43	21,36	21,77	21,96
	6	21,28	21,68	23,92	24,29	25,24	21,43	21,43	22,02	22,22
rze	5	24,11	24,48	26,72	26,78	25,24	25,24	25,24	26,40	26,73
dok	3	24,84	24,84	24,91	24,84	26,78	24,29	24,15	23,51	23,75
wie	2	24,43	24,43	24,43	24,91	26,72	23,92	23,81	23,16	23,44
2.	1	20,56	20,56	24,43	24,84	24,48	21,68	21,58	20,90	21,14
	0	20,01	20,56	24,43	24,84	24,11	21,28	21,15	20,38	20,60

Tabela A12. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

					1. wide	ok rzeczy	wisty			
		0	1	2	3	5	6	7	8	9
	9	18,08	18,81	21,86	22,34	24,11	20,29	19,91	18,03	17,80
wisty	8	17,87	18,64	21,50	22,03	23,20	19,96	19,64	17,28	18,03
	7	19,16	19,61	22,36	22,85	23,07	20,30	19,71	19,64	19,91
czy	6	19,23	19,59	21,91	22,31	22,07	19,82	20,30	19,96	20,29
rze	5	23,66	23,33	25,31	25,43	24,21	22,07	23,07	23,20	24,11
dok	3	21,88	21,95	22,41	22,25	25,43	22,31	22,85	22,03	22,34
2. wid	2	20,99	21,16	21,41	22,41	25,31	21,91	22,36	21,50	21,86
	1	17,48	17,59	21,16	21,95	23,33	19,59	19,61	18,64	18,81
	0	16,67	17,48	20,99	21,88	23,66	19,23	19,16	17,87	18,08

BALLET

Tabela A13. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

				1. wie	lok rzeczy	wisty		
		0	1	2	3	5	6	7
ý	7	24,68	25,58	26,37	28,46	32,54	27,38	23,76
wist	6	25,83	26,43	27,68	29,10	32,54	26,54	27,38
czyv	5	28,47	29,51	30,53	32,16	32,54	32,54	32,54
rze	3	29,59	29,59	30,47	29,59	32,16	29,10	28,46
dok	2	26,45	26,45	26,45	30,47	30,53	27,68	26,37
wiu	1	22,85	22,70	26,45	29,59	29,51	26,43	25,58
5.	0	20,97	22,85	26,45	29,59	28,47	25,83	24,68

Tabela A14. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

				1. wie	dok rzeczy	wisty		
		0	1	2	3	5	6	7
ý	7	21,79	22,76	24,70	27,73	26,74	23,55	21,00
wist	6	23,45	24,26	25,94	28,29	26,75	22,84	23,55
czy	5	27,33	27,51	29,71	31,32	26,28	26,75	26,74
rze	3	24,60	25,87	26,28	25,51	31,32	28,29	27,73
dok	2	23,16	23,64	22,53	26,28	29,71	25,94	24,70
wic	1	19,88	19,63	23,64	25,87	27,51	24,26	22,76
5	0	18,68	19,88	23,16	24,60	27,33	23,45	21,79

BREAKDANCERS

Tabela A15. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

				1. wie	lok rzeczy	wisty		
		0	1	2	3	5	6	7
y	7	24,89	25,88	28,07	30,80	31,99	28,21	24,59
wist	6	27,40	27,58	28,77	31,12	31,99	28,21	28,21
czy	5	30,05	30,24	31,49	32,07	31,99	31,99	31,99
rze	3	30,36	30,36	30,62	30,36	32,07	31,12	30,80
dok	2	27,11	27,11	27,11	30,62	31,49	28,77	28,07
wid	1	22,40	22,40	27,11	30,36	30,24	27,58	25,88
5.	0	20,69	22,40	27,11	30,36	30,05	27,40	24,89

Tabela A16. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

			1. widok rzeczywisty									
		0	1	2	3	5	6	7				
ý	7	22,70	24,00	26,67	30,31	29,82	24,73	21,80				
czywist	6	25,44	25,83	27,82	30,56	30,30	25,32	24,73				
	5	29,14	29,12	31,03	31,41	30,87	30,30	29,82				
rze	3	27,92	27,53	28,63	28,76	31,41	30,56	30,31				
dok	2	23,54	23,85	25,37	28,63	31,03	27,82	26,67				
wie	1	19,83	20,14	23,85	27,53	29,12	25,83	24,00				
5	0	17,83	19,83	23,54	27,92	29,14	25,44	22,70				

SOCCER ARC

Tabela A17. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

		1. widok rzeczywisty								
		1	2	3	5	6	7			
	7	21,24	22,16	22,73	21,85	20,38	20,83			
ty .	6	20,71	21,71	22,22	21,76	20,13	20,38			
idok wis	5	20,36	22,19	22,71	20,41	21,76	21,85			
. wi eczy	3	23,12	23,27	23,12	22,71	22,22	22,73			
rze	2	22,07	22,07	23,27	22,19	21,71	22,16			
	1	19,36	22,07	23,12	20,36	20,71	21,24			

Tabela A18. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

			1. widok rzeczywisty							
		1	2	3	5	6	7			
	7	19,49	20,87	21,93	18,77	18,04	17,88			
ty	6	19,79	21,13	22,08	18,80	17,69	18,04			
dok wis	5	19,42	20,64	22,39	17,75	18,80	18,77			
. wi eczy	3	21,33	21,29	20,98	22,39	22,08	21,93			
2 rze	2	19,40	19,70	21,29	20,64	21,13	20,87			
	1	17,44	19,40	21,33	19,42	19,79	19,49			

SOCCER LINEAR

Tabela A19. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

					1. widok r	zeczywisty		
			1	2	3	5	6	7
		7	29,56	30,36	32,35	32,18	29,09	27,34
t L	, [6	30,37	31,16	32,94	32,18	29,09	29,09
idok		5	32,37	33,01	34,62	32,18	32,18	32,18
. wi eczy		3	32,28	32,28	32,28	34,62	32,94	32,35
rze		2	30,14	30,14	32,28	33,01	31,16	30,36
		1	29,02	30,14	32,28	32,37	30,37	29,56

Tabela A20. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

			1. widok rzeczywisty						
		1	2	3	5	6	7		
	7	29,61	30,42	32,82	30,99	28,31	26,70		
ty	6	30,59	31,21	33,16	31,01	28,46	28,31		
idok wis	5	32,78	33,18	34,53	31,56	31,01	30,99		
wi eczy	3	31,04	31,21	31,36	34,53	33,16	32,82		
rze 2	2	27,24	27,32	31,21	33,18	31,21	30,42		
	1	26,08	27,24	31,04	32,78	30,59	29,61		

POZNAN_CARPARK

Tabela A21. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupelnianych)

			1. widok rzeczywisty							
		1	2	3	5	6	7			
	7	28,93	30,22	31,85	32,73	29,97	26,86			
ty .	6	30,35	31,37	32,92	32,73	29,97	29,97			
idok wis	5	31,42	32,71	34,81	32,73	32,73	32,73			
czy	3	33,74	33,74	33,74	34,81	32,92	31,85			
126	2	30,30	30,30	33,74	32,71	31,37	30,22			
	1	28,05	30,30	33,74	31,42	30,35	28,93			

Tabela A22. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

			1. widok rzeczywisty						
		1	2	3	5	6	7		
	7	27,56	28,87	30,53	28,55	25,83	23,37		
ty	6	29,01	30,09	31,71	29,67	26,26	25,83		
idok wis	5	30,50	31,94	33,89	29,91	29,67	28,55		
eczy	3	30,30	30,35	30,52	33,89	31,71	30,53		
rze	2	27,00	27,50	30,35	31,94	30,09	28,87		
	1	24,94	27,00	30,30	30,50	29,01	27,56		

POZNAN_STREET

Tabela A23. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone wyłącznie dla obszarów przerzutowanych z widoków rzeczywistych (nie uwzględniono obszarów uzupełnianych)

		1. widok rzeczywisty					
		1	2	3	5	6	7
	7	27,44	28,46	29,24	34,32	30,41	26,18
dok wisty	6	30,47	31,76	33,15	34,32	30,41	30,41
	5	33,19	34,03	35,55	34,32	34,32	34,32
. wi eczy	3	33,01	33,01	33,01	35,55	33,15	29,24
126	2	31,03	31,03	33,01	34,03	31,76	28,46
	1	28,65	31,03	33,01	33,19	30,47	27,44

Tabela A24. Jakość widoku wirtualnego 4 syntezowanego z różnych par widoków rzeczywistych; wartości PSNR [dB] wyznaczone dla całego widoku wirtualnego

		1. widok rzeczywisty					
		1	2	3	5	6	7
	7	27,55	29,20	31,86	29,49	26,65	23,58
2. widok rzeczywisty	6	29,89	31,20	33,44	30,07	27,50	26,65
	5	33,01	33,87	35,86	30,56	30,07	29,49
	3	31,91	32,34	31,79	35,86	33,44	31,86
	2	29,18	29,62	32,34	33,87	31,20	29,20
	1	27,00	29,18	31,91	33,01	29,89	27,55

A2. WPŁYW FILTRACJI NA JAKOŚĆ SYNTEZY

Patrz: rozdział 6.3.

BBB BUTTERFLY

	PSNR [dB]				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
19	29,27	32,35	32,83		
32	30,07	32,86	33,00		
45	31,07	33,41	33,70		
58	30,99	32,93	33,71		
71	31,74	33,10	34,67		
Średnio	30,63	32,93	33,58		

Tabela A25. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A26. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
19	0,915	0,949	0,959		
32	0,927	0,954	0,959		
45	0,936	0,962	0,967		
58	0,942	0,963	0,968		
71	0,949	0,965	0,972		
Średnio	0,934	0,959	0,965		

BBB FLOWERS

	PSNR [dB]				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
19	23,83	24,99	25,12		
32	25,38	26,87	26,87		
45	23,96	25,78	26,26		
58	23,72	24,95	24,94		
71	24,94	26,60	26,69		
Średnio	24,36	25,84	25,97		

Tabela A27. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A28. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
19	0,789	0,843	0,850		
32	0,814	0,869	0,874		
45	0,830	0,877	0,884		
58	0,822	0,880	0,887		
71	0,837	0,891	0,898		
Średnio	0,818	0,872	0,878		

POZNAN_BLOCKS

		Ç ,			
	PSNR [dB]				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
1	22,63	22,94	22,98		
2	23,32	23,72	23,75		
3	24,39	25,02	25,02		
4	25,20	25,75	25,76		
5	24,43	25,04	25,08		
6	25,95	26,53	26,32		
7	25,84	26,55	26,61		
8	25,67	26,07	26,17		
Średnio	24,68	25,20	25,21		

Tabela A29. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A30. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
1	0,684	0,717	0,734		
2	0,713	0,755	0,769		
3	0,721	0,761	0,770		
4	0,767	0,802	0,811		
5	0,762	0,807	0,814		
6	0,787	0,821	0,827		
7	0,776	0,812	0,822		
8	0,781	0,815	0,827		
Średnio	0,749	0,786	0,797		

POZNAN_BLOCKS2

	PSNR [dB]				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
1	33,28	33,47	33,68		
2	33,05	33,65	33,68		
3	31,05	31,27	31,33		
4	30,03	30,22	30,23		
5	30,25	30,36	30,37		
6	30,60	30,88	30,89		
7	30,35	30,54	30,61		
8	30,96	31,80	32,14		
Średnio	31,20	31,52	31,62		

Tabela A31. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A32. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
1	0,881	0,887	0,891		
2	0,872	0,881	0,883		
3	0,879	0,888	0,890		
4	0,845	0,851	0,852		
5	0,821	0,832	0,833		
6	0,828	0,832	0,832		
7	0,826	0,834	0,836		
8	0,855	0,868	0,871		
Średnio	0,851	0,859	0,861		

POZNAN_FENCING2

	PSNR [dB]				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
1	30,33	30,52	30,67		
2	33,32	33,54	33,59		
3	31,58	32,20	32,21		
4	28,12	28,24	28,23		
5	27,82	27,72	27,77		
6	29,96	30,04	30,05		
7	30,84	30,56	30,61		
8	32,01	32,32	32,35		
Średnio	30,50	30,64	30,69		

Tabela A33. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A34. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
1	0,886	0,893	0,895		
2	0,900	0,904	0,905		
3	0,884	0,895	0,895		
4	0,851	0,856	0,857		
5	0,845	0,847	0,848		
6	0,864	0,867	0,868		
7	0,868	0,872	0,873		
8	0,892	0,903	0,904		
Średnio	0,874	0,880	0,880		

POZNAN_SERVICE2

	PSNR [dB]		
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
1	19,19	19,61	19,77
2	24,82	24,81	24,94
3	27,67	27,99	28,23
4	26,43	26,79	26,97
5	26,48	26,42	26,64
6	25,52	25,55	25,55
7	25,36	25,18	25,40
8	26,26	25,70	26,38
Średnio	25,22	25,26	25,48

Tabela A35. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A36. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Widok	SSIM		
	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
1	0,752	0,767	0,771
2	0,804	0,816	0,819
3	0,850	0,861	0,864
4	0,803	0,816	0,819
5	0,815	0,829	0,831
6	0,795	0,802	0,804
7	0,796	0,806	0,810
8	0,800	0,807	0,812
Średnio	0,802	0,813	0,816

BALLET

Widok	PSNR [dB]		
	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
1	25,79	26,03	26,52
2	27,18	27,24	27,29
3	31,22	31,27	31,35
4	32,56	32,67	32,74
5	32,35	32,52	32,61
6	29,83	29,96	30,03
Średnio	29,82	29,95	30,09

Tabela A37. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A38. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM		
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
1	0,770	0,779	0,796
2	0,786	0,789	0,795
3	0,874	0,881	0,885
4	0,877	0,881	0,885
5	0,877	0,883	0,887
6	0,856	0,865	0,869
Średnio	0,840	0,846	0,853

BREAKDANCERS

	PSNR [dB]		
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
1	28,46	28,76	29,03
2	31,08	31,48	31,49
3	31,41	31,59	31,60
4	31,60	31,90	32,07
5	33,52	33,59	33,66
6	31,30	31,54	31,93
Średnio	31,23	31,48	31,63

Tabela A39. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A40. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM		
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
1	0,805	0,814	0,815
2	0,831	0,838	0,839
3	0,846	0,852	0,853
4	0,856	0,863	0,864
5	0,858	0,862	0,864
6	0,830	0,840	0,841
Średnio	0,838	0,845	0,846

SOCCER ARC

Widok	PSNR [dB]		
	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
2	22,07	22,35	22,54
3	22,29	22,49	22,51
4	23,10	23,29	23,36
5	19,16	19,26	19,27
6	18,20	18,36	18,37
Średnio	20,96	21,15	21,21

Tabela A41. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A42. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Widok	SSIM		
	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
2	0,766	0,786	0,787
3	0,807	0,823	0,824
4	0,803	0,827	0,827
5	0,673	0,677	0,678
6	0,649	0,659	0,660
Średnio	0,740	0,755	0,755

SOCCER LINEAR

Widok	PSNR [dB]		
	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
2	33,75	33,73	33,65
3	33,92	33,89	33,85
4	34,68	34,71	34,63
5	35,39	35,55	35,47
6	33,87	33,89	33,79
Średnio	34,32	34,35	34,28

Tabela A43. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A44. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Widok	SSIM		
	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
2	0,891	0,891	0,891
3	0,909	0,910	0,911
4	0,916	0,916	0,916
5	0,922	0,922	0,923
6	0,887	0,887	0,888
Średnio	0,905	0,905	0,906

POZNAN_CARPARK

	BSNIP [4B]		
Widok			r
	Brak filtracji	Wypełnianie	Pełna filtracja
	,	pęknięć	
2	35,34	35,57	35,43
3	33,49	33,47	33,37
4	34,04	34,22	33,99
5	33,15	33,49	33,26
6	33,36	33,45	33,34
Średnio	33,88	34,04	33,88

Tabela A45. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A46. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM		
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja
2	0,941	0,941	0,941
3	0,932	0,932	0,932
4	0,932	0,932	0,932
5	0,928	0,930	0,929
6	0,928	0,928	0,928
Średnio	0,932	0,933	0,932

POZNAN_STREET

	PSNR [dB]				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
2	35,92	35,91	35,86		
3	34,61	34,65	34,58		
4	35,85	35,86	35,81		
5	36,53	36,41	36,33		
6	36,23	36,09	36,01		
Średnio	35,83	35,79	35,72		

Tabela A47. Jakość syntezowanych widoków dla trzech konfiguracji MVS

Tabela A48. Jakość syntezowanych widoków dla trzech konfiguracji MVS

	SSIM				
Widok	Brak filtracji	Wypełnianie pęknięć	Pełna filtracja		
2	0,940	0,940	0,940		
3	0,933	0,933	0,932		
4	0,937	0,937	0,937		
5	0,941	0,941	0,941		
6	0,938	0,937	0,937		
Średnio	0,938	0,938	0,937		

A3. RZUTOWANIE ZAPEWNIAJĄCE CIĄGŁOŚĆ OBIEKTÓW

Patrz: rozdział 7.3.

BBB BUTTERFLY

Tabela A49. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNR [dB]		SSIM	
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów
19	31,01	32,83	0,943	0,959
32	32,17	33,00	0,947	0,959
45	32,65	33,70	0,955	0,967
58	32,82	33,71	0,958	0,968
71	33,61	34,67	0,964	0,972
Średnio	32,45	33,58	0,954	0,965

BBB FLOWERS

Tabela A50. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNR [dB]		SSIM	
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów
19	24,81	25,12	0,845	0,850
32	26,50	26,87	0,867	0,874
45	25,90	26,26	0,878	0,884
58	24,86	24,94	0,882	0,887
71	26,32	26,69	0,889	0,898
Średnio	25,68	25,97	0,872	0,878

POZNAN_BLOCKS

	PSNI	R [dB]	SSIM	
		Rzutowanie		Rzutowanie
Widok	Rzutowanie	zapewniające	Rzutowanie	zapewniające
	niezależne	ciągłość	niezależne	ciągłość
		obiektów		obiektów
1	22,96	22,98	0,733	0,734
2	23,73	23,75	0,767	0,769
3	24,90	25,02	0,767	0,770
4	25,71	25,76	0,810	0,811
5	25,04	25,08	0,812	0,814
6	26,24	26,32	0,826	0,827
7	26,53	26,61	0,819	0,822
8	26,11	26,17	0,825	0,827
Średnio	25,15	25,21	0,795	0,797

Tabela A51. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

POZNAN_BLOCKS2

Tabela A52. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNI	R [dB]	SSIM	
		Rzutowanie		Rzutowanie
Widok	Rzutowanie	zapewniające	Rzutowanie	zapewniające
	niezależne	ciągłość	niezależne	ciągłość
		obiektów		obiektów
1	33,80	33,68	0,893	0,891
2	33,61	33,68	0,883	0,883
3	31,32	31,33	0,890	0,890
4	30,21	30,23	0,853	0,852
5	30,31	30,37	0,832	0,833
6	30,91	30,89	0,833	0,832
7	30,59	30,61	0,836	0,836
8	31,86	32,14	0,871	0,871
Średnio	31,58	31,62	0,861	0,861

POZNAN_FENCING2

	PSNI	R [dB]	SS	IM
		Rzutowanie		Rzutowanie
Widok	Rzutowanie	zapewniające	Rzutowanie	zapewniające
	niezależne	ciągłość	niezależne	ciągłość
		obiektów		obiektów
1	30,65	30,67	0,895	0,895
2	33,57	33,59	0,905	0,905
3	32,18	32,21	0,894	0,895
4	28,24	28,23	0,859	0,857
5	27,76	27,77	0,848	0,848
6	30,04	30,05	0,868	0,868
7	30,60	30,61	0,873	0,873
8	32,35	32,35	0,904	0,904
Średnio	30,67	30,69	0,881	0,880

Tabela A53. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

POZNAN_SERVICE2

Tabela A54. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNI	R [dB]	SSIM	
		Rzutowanie		Rzutowanie
Widok	Rzutowanie	zapewniające	Rzutowanie	zapewniające
	niezależne	ciągłość	niezależne	ciągłość
		obiektów		obiektów
1	19,64	19,77	0,770	0,771
2	24,88	24,94	0,818	0,819
3	28,21	28,23	0,863	0,864
4	26,89	26,97	0,822	0,819
5	26,63	26,64	0,831	0,831
6	25,50	25,55	0,803	0,804
7	25,38	25,40	0,810	0,810
8	26,31	26,38	0,811	0,812
Średnio	25,43	25,48	0,816	0,816

BALLET

	PSNR [dB]		SSIM	
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów
1	26,46	26,52	0,794	0,796
2	27,26	27,29	0,795	0,795
3	31,39	31,35	0,887	0,885
4	32,74	32,74	0,887	0,885
5	32,40	32,61	0,890	0,887
6	30,00	30,03	0,868	0,869
Średnio	30,04	30,09	0,854	0,853

Tabela A55. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

BREAKDANCERS

Tabela A56. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNR [dB]		SSIM	
		Rzutowanie		Rzutowanie
Widok	Rzutowanie	zapewniające	Rzutowanie	zapewniające
	niezależne	ciągłość	niezależne	ciągłość
		obiektów		obiektów
1	28,97	29,03	0,815	0,815
2	31,50	31,49	0,839	0,839
3	31,59	31,60	0,852	0,853
4	32,09	32,07	0,864	0,864
5	33,66	33,66	0,863	0,864
6	31,93	31,93	0,841	0,841
Średnio	31,62	31,63	0,846	0,846

SOCCER ARC

	PSNR [dB]		SSIM				
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów			
2	22,50	22,54	0,788	0,787			
3	22,48	22,51	0,824	0,824			
4	23,18	23,36	0,832	0,827			
5	19,21	19,27	0,677	0,678			
6	18,34	18,37	0,660	0,660			
Średnio	21,14	21,21	0,756	0,755			

Tabela A57. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

SOCCER LINEAR

Tabela A58. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNR [dB]		SSIM	
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów
2	33,60	33,65	0,891	0,891
3	33,81	33,85	0,911	0,911
4	34,61	34,63	0,916	0,916
5	35,41	35,47	0,923	0,923
6	33,75	33,79	0,887	0,888
Średnio	34,23	34,28	0,906	0,906

POZNAN_CARPARK

	PSNR [dB]		SSIM	
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów
2	35,39	35,43	0,941	0,941
3	33,35	33,37	0,932	0,932
4	33,95	33,99	0,932	0,932
5	33,26	33,26	0,928	0,929
6	33,32	33,34	0,928	0,928
Średnio	33,85	33,88	0,932	0,932

Tabela A59. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

POZNAN_STREET

Tabela A60. Porównanie jakości widoków wirtualnych dla niezależnego rzutowania punktów i rzutowania zapewniającego ciągłość obiektów

	PSNR [dB]		SSIM	
Widok	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów	Rzutowanie niezależne	Rzutowanie zapewniające ciągłość obiektów
2	35,85	35,86	0,940	0,940
3	34,58	34,58	0,932	0,932
4	35,80	35,81	0,937	0,937
5	36,32	36,33	0,941	0,941
6	36,00	36,01	0,937	0,937
Średnio	35,71	35,72	0,937	0,937

A4. KOREKCJA NIESPÓJNOŚCI BARWNEJ

Patrz: rozdział 8.4.

BBB BUTTERFLY

	PSNR [dB]				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
19	32,81	32,81	32,83	32,83	
32	33,01	33,01	33,00	33,00	
45	33,75	33,75	33,70	33,70	
58	33,72	33,72	33,71	33,71	
71	34,71	34,71	34,67	34,67	
Średnio	33,60	33,60	33,58	33,58	

Tabela A61. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A62. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
19	0,959	0,959	0,959	0,959
32	0,959	0,959	0,959	0,959
45	0,967	0,967	0,967	0,967
58	0,968	0,968	0,968	0,968
71	0,972	0,972	0,972	0,972
Średnio	0,965	0,965	0,965	0,965

BBB FLOWERS

	PSNR [dB]			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
19	25,13	25,13	25,12	25,12
32	26,90	26,90	26,87	26,87
45	26,24	26,24	26,26	26,26
58	24,92	24,92	24,94	24,94
71	26,79	26,79	26,69	26,69
Średnio	25,99	25,99	25,97	25,97

Tabela A63. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A64. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
19	0,849	0,849	0,850	0,850
32	0,873	0,873	0,874	0,874
45	0,884	0,884	0,884	0,884
58	0,886	0,886	0,887	0,887
71	0,898	0,898	0,898	0,898
Średnio	0,878	0,878	0,878	0,878

POZNAN_BLOCKS

	DENID [4D]				
Widok	roink [ab]				
	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	23,00	23,00	22,92	22,98	
2	23,74	23,74	23,74	23,75	
3	24,93	24,90	25,02	25,02	
4	25,75	25,69	25,79	25,76	
5	25,00	25,00	25,08	25,08	
6	26,26	26,30	26,30	26,32	
7	26,50	26,50	26,62	26,61	
8	26,19	26,17	26,19	26,17	
Średnio	25,17	25,16	25,21	25,21	

Tabela A65. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A66. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	0,732	0,732	0,733	0,734	
2	0,766	0,766	0,769	0,769	
3	0,766	0,768	0,770	0,770	
4	0,807	0,808	0,811	0,811	
5	0,812	0,812	0,814	0,814	
6	0,825	0,825	0,827	0,827	
7	0,819	0,819	0,822	0,822	
8	0,825	0,825	0,827	0,827	
Średnio	0,794	0,794	0,797	0,797	

POZNAN_BLOCKS2

	PSNR [dB]				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	33,61	33,71	33,58	33,68	
2	33,53	33,63	33,66	33,68	
3	31,21	31,31	31,22	31,33	
4	28,03	30,16	28,13	30,23	
5	27,91	30,36	27,89	30,37	
6	30,87	30,91	30,86	30,89	
7	30,63	30,59	30,65	30,61	
8	31,80	31,90	31,94	32,14	
Średnio	30,95	31,57	30,99	31,62	

Tabela A67. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A68. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	0,890	0,890	0,890	0,891	
2	0,881	0,882	0,882	0,883	
3	0,888	0,889	0,889	0,890	
4	0,849	0,852	0,850	0,852	
5	0,830	0,833	0,831	0,833	
6	0,832	0,832	0,832	0,832	
7	0,836	0,836	0,837	0,836	
8	0,869	0,870	0,871	0,871	
Średnio	0,859	0,860	0,860	0,861	
POZNAN_FENCING2

	PSNR [dB]				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	30,64	30,65	30,67	30,67	
2	33,41	33,50	33,55	33,59	
3	32,12	32,17	32,17	32,21	
4	25,63	28,23	25,63	28,23	
5	24,77	27,76	24,70	27,77	
6	29,99	30,03	30,02	30,05	
7	30,47	30,58	30,51	30,61	
8	31,83	32,22	31,83	32,35	
Średnio	29,86	30,64	29,88	30,69	

Tabela A69. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A70. Jakość syntezowanych widoków dla czterech konfiguracji MVS

		SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	0,891	0,893	0,893	0,895	
2	0,900	0,903	0,903	0,905	
3	0,893	0,893	0,894	0,895	
4	0,847	0,855	0,851	0,857	
5	0,835	0,847	0,840	0,848	
6	0,866	0,866	0,867	0,868	
7	0,868	0,872	0,870	0,873	
8	0,896	0,902	0,899	0,904	
Średnio	0,874	0,879	0,877	0,880	

POZNAN_SERVICE2

		PSNF	PSNR [dB]			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz		
1	19,74	19,78	19,73	19,77		
2	24,93	24,93	24,94	24,94		
3	28,20	28,19	28,24	28,23		
4	25,71	26,94	25,65	26,97		
5	25,41	26,65	25,26	26,64		
6	25,39	25,53	25,41	25,55		
7	25,18	25,33	25,28	25,40		
8	26,20	26,38	26,11	26,38		
Średnio	25,09	25,47	25,08	25,48		

Tabela A71. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A72. Jakość syntezowanych widoków dla czterech konfiguracji MVS

		SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	0,769	0,770	0,770	0,771	
2	0,818	0,819	0,819	0,819	
3	0,862	0,862	0,863	0,864	
4	0,810	0,818	0,816	0,819	
5	0,815	0,824	0,821	0,831	
6	0,802	0,803	0,804	0,804	
7	0,809	0,812	0,805	0,810	
8	0,808	0,811	0,811	0,812	
Średnio	0,812	0,815	0,814	0,816	

BALLET

	PSNR [dB]				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
1	26,51	26,81	26,18	26,52	
2	27,05	27,15	27,22	27,29	
3	31,40	31,71	30,99	31,35	
4	32,06	32,56	32,46	32,74	
5	32,46	32,65	32,88	32,61	
6	30,79	30,79	30,03	30,03	
Średnio	30,05	30,28	29,96	30,09	

Tabela A73. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A74. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
1	0,815	0,815	0,796	0,796
2	0,793	0,793	0,795	0,795
3	0,883	0,883	0,885	0,885
4	0,883	0,884	0,885	0,885
5	0,884	0,886	0,887	0,887
6	0,876	0,876	0,869	0,869
Średnio	0,856	0,856	0,853	0,853

BREAKDANCERS

	PSNR [dB]			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
1	29,71	29,81	28,84	29,03
2	31,32	31,32	31,49	31,49
3	31,80	31,81	31,59	31,60
4	32,09	32,09	32,07	32,07
5	33,68	33,87	33,41	33,66
6	33,00	33,43	31,25	31,93
Średnio	31,94	32,06	31,44	31,63

Tabela A75. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A76. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
1	0,843	0,844	0,813	0,815
2	0,837	0,837	0,839	0,839
3	0,853	0,853	0,853	0,853
4	0,864	0,864	0,864	0,864
5	0,863	0,863	0,862	0,864
6	0,876	0,878	0,837	0,841
Średnio	0,856	0,857	0,845	0,846

SOCCER ARC

	PSNR [dB]				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz	
2	22,67	22,33	23,21	22,54	
3	21,97	22,15	22,35	22,51	
4	22,70	23,13	23,24	23,36	
5	18,93	19,32	18,83	19,27	
6	18,08	18,14	18,66	18,37	
Średnio	20,87	21,02	21,26	21,21	

Tabela A77. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A78. Jakość syntezowanych widoków dla czterech konfiguracji MVS

		SSIM				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz		
2	0,791	0,785	0,798	0,787		
3	0,816	0,819	0,822	0,824		
4	0,806	0,825	0,833	0,827		
5	0,665	0,673	0,666	0,678		
6	0,660	0,658	0,670	0,660		
Średnio	0,747	0,752	0,758	0,755		

SOCCER LINEAR

	PSNR [dB]			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
2	33,65	33,65	33,65	33,65
3	33,87	33,87	33,85	33,85
4	34,64	34,64	34,63	34,63
5	35,44	35,44	35,47	35,47
6	33,79	33,79	33,79	33,79
Średnio	34,28	34,28	34,28	34,28

Tabela A79. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A80. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
2	0,891	0,891	0,891	0,891
3	0,911	0,911	0,911	0,911
4	0,916	0,916	0,916	0,916
5	0,923	0,923	0,923	0,923
6	0,888	0,888	0,888	0,888
Średnio	0,906	0,906	0,906	0,906

POZNAN_CARPARK

	PSNR [dB]			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
2	35,43	35,43	35,43	35,43
3	33,42	33,42	33,37	33,37
4	33,97	33,97	33,99	33,99
5	33,31	33,31	33,26	33,26
6	33,36	33,36	33,34	33,34
Średnio	33,90	33,90	33,88	33,88

Tabela A81. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A82. Jakość syntezowanych widoków dla czterech konfiguracji MVS

	SSIM			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
2	0,941	0,941	0,941	0,941
3	0,933	0,933	0,932	0,932
4	0,931	0,931	0,932	0,932
5	0,931	0,931	0,929	0,929
6	0,928	0,928	0,928	0,928
Średnio	0,933	0,933	0,932	0,932

POZNAN_STREET

	PSNR [dB]			
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz
2	35,91	35,91	35,86	35,86
3	36,10	36,10	34,58	34,58
4	36,43	36,43	35,81	35,81
5	37,44	37,44	36,33	36,33
6	36,18	36,18	36,01	36,01
Średnio	36,41	36,41	35,72	35,72

Tabela A83. Jakość syntezowanych widoków dla czterech konfiguracji MVS

Tabela A84. Jakość syntezowanych widoków dla czterech konfiguracji MVS

		SSIM				
Widok	Brak korekcji niespójności barwnej	Szybka korekcja	Adaptacyjna korekcja	Obie korekcje niespójności barwnej naraz		
2	0,940	0,940	0,940	0,940		
3	0,933	0,933	0,932	0,932		
4	0,936	0,936	0,937	0,937		
5	0,941	0,941	0,941	0,941		
6	0,938	0,938	0,937	0,937		
Średnio	0,938	0,938	0,937	0,937		

A5. WSTĘPNE ZWIĘKSZANIE ROZDZIELCZOŚCI

Patrz: rozdział 9.3.

BBB BUTTERFLY

	PSNF	PSNR [dB]		IM
Widok	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości
19	32,14	32,83	0,947	0,959
32	32,68	33,00	0,955	0,959
45	32,99	33,70	0,957	0,967
58	33,01	33,71	0,955	0,968
71	34,10	34,67	0,966	0,972
Średnio	32,98	33,58	0,956	0,965

Tabela A85. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

BBB FLOWERS

	PSNF	NR [dB] SS		IM
Widok	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości
19	24,48	25,12	0,836	0,850
32	26,08	26,87	0,860	0,874
45	25,41	26,26	0,869	0,884
58	24,29	24,94	0,868	0,887
71	25,77	26,69	0,879	0,898
Średnio	25,21	25,97	0,863	0,878

Tabela A86. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

POZNAN_BLOCKS

	PSNF	R [dB]	SSIM	
W/: -1 - 1-	Brak	Wstępne	Brak	Wstępne
WICOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
1	22,81	22,98	0,729	0,734
2	23,47	23,75	0,762	0,769
3	24,90	25,02	0,769	0,770
4	25,66	25,76	0,802	0,811
5	24,95	25,08	0,799	0,814
6	26,09	26,32	0,817	0,827
7	26,34	26,61	0,804	0,822
8	25,93	26,17	0,808	0,827
Średnio	25,02	25,21	0,786	0,797

Tabela A87. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

POZNAN_BLOCKS2

Tabela A88. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

	PSNF	R [dB]	SSIM	
Widok	Brak	Wstępne	Brak	Wstępne
WIGON	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
1	33,18	33,68	0,887	0,891
2	33,01	33,68	0,879	0,883
3	30,80	31,33	0,888	0,890
4	29,48	30,23	0,851	0,852
5	29,74	30,37	0,827	0,833
6	30,36	30,89	0,834	0,832
7	29,86	30,61	0,828	0,836
8	31,57	32,14	0,870	0,871
Średnio	31,00	31,62	0,858	0,861

POZNAN_FENCING2

	PSNF	R [dB]	SS	IM
Widol	Brak	Wstępne	Brak	Wstępne
WICOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
1	29,97	30,67	0,872	0,895
2	33,01	33,59	0,890	0,905
3	31,53	32,21	0,890	0,895
4	27,66	28,23	0,849	0,857
5	27,17	27,77	0,847	0,848
6	29,52	30,05	0,868	0,868
7	30,17	30,61	0,866	0,873
8	31,79	32,35	0,899	0,904
Średnio	30,10	30,69	0,873	0,880

Tabela A89. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

POZNAN_SERVICE2

Tabela A90.	Jakość syntezov	wanych widoków	dla dwóch l	konfiguracji MVS
	J	2		0 /

	PSNF	R [dB]	SSIM	
W/: J _ 1_	Brak	Wstępne	Brak	Wstępne
WICOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
1	19,47	19,77	0,783	0,771
2	24,64	24,94	0,816	0,819
3	27,84	28,23	0,860	0,864
4	26,74	26,97	0,804	0,819
5	26,37	26,64	0,816	0,831
6	25,18	25,55	0,778	0,804
7	25,06	25,40	0,805	0,810
8	26,05	26,38	0,800	0,812
Średnio	25,17	25,48	0,808	0,816

BALLET

	PSNF	R [dB]	SSIM	
Widala	Brak	Wstępne	Brak	Wstępne
WIGOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
1	26,23	26,52	0,788	0,796
2	26,81	27,29	0,790	0,795
3	30,93	31,35	0,878	0,885
4	32,28	32,74	0,870	0,885
5	32,20	32,61	0,874	0,887
6	29,49	30,03	0,849	0,869
Średnio	29,66	30,09	0,841	0,853

Tabela A91. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

BREAKDANCERS

Tabela A92.	Jakość syntezowanyc	h widoków dla	dwóch konf	iguracji MVS
				0 /

	PSNF	R [dB]	SSIM	
Widala	Brak	Wstępne	Brak	Wstępne
WIGOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
1	28,81	29,03	0,799	0,815
2	31,19	31,49	0,831	0,839
3	31,17	31,60	0,847	0,853
4	31,74	32,07	0,851	0,864
5	33,29	33,66	0,834	0,864
6	31,53	31,93	0,813	0,841
Średnio	31,29	31,63	0,829	0,846

SOCCER ARC

	PSNF	R [dB]	SSIM		
Widok	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości	Brak zwiększania rozdzielczości	Wstępne zwiększanie rozdzielczości	
2	22,25	22,54	0,766	0,787	
3	22,06	22,51	0,806	0,824	
4	22,83	23,36	0,805	0,827	
5	19,10	19,27	0,661	0,678	
6	18,14	18,37	0,630	0,660	
Średnio	20,88	21,21	0,733	0,755	

Tabela A93. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

SOCCER LINEAR

Tabela A94. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

	PSNF	R [dB]	SSIM	
Widok	Brak	Wstępne	Brak	Wstępne
WIGOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
2	33,53	33,65	0,891	0,891
3	33,67	33,85	0,910	0,911
4	34,72	34,63	0,921	0,916
5	35,31	35,47	0,920	0,923
6	33,66	33,79	0,886	0,888
Średnio	34,18	34,28	0,906	0,906

POZNAN_CARPARK

	PSNF	R [dB]	SSIM	
Widok	Brak	Wstępne	Brak	Wstępne
WIGOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
2	35,32	35,43	0,947	0,941
3	33,17	33,37	0,930	0,932
4	33,68	33,99	0,937	0,932
5	32,97	33,26	0,922	0,929
6	33,06	33,34	0,931	0,928
Średnio	33,64	33,88	0,933	0,932

Tabela A95. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

POZNAN_STREET

Tabela A96. Jakość syntezowanych widoków dla dwóch konfiguracji MVS

	PSNF	R [dB]	SSIM	
Widok	Brak	Wstępne	Brak	Wstępne
WIGOK	zwiększania	zwiększanie	zwiększania	zwiększanie
	rozdzielczości	rozdzielczości	rozdzielczości	rozdzielczości
2	35,57	35,86	0,930	0,940
3	34,34	34,58	0,941	0,932
4	35,53	35,81	0,936	0,937
5	36,19	36,33	0,937	0,941
6	35,80	36,01	0,932	0,937
Średnio	35,49	35,72	0,935	0,937

A6. Ocena efektywności zaproponowanej metody syntezy

Patrz: rozdział 10.2.1.

BBB BUTTERFLY

		PSNR [dB]			
Widok	WEDE	MVS	MVS	MVS	
	VSK5	(2 widoki)	(4 widoki)	(wszystkie widoki)	
19	28,99	31,65	32,65	32,83	
32	31,07	31,97	32,76	33,00	
45	31,95	32,58	33,43	33,70	
58	31,89	32,70	33,52	33,71	
71	32,36	33,69	34,49	34,67	
Średnio	31,25	32,52	33,37	33,58	

Tabela A97. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A98. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSBS	MVS	MVS	MVS	
	V SKS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
19	0,937	0,956	0,959	0,959	
32	0,947	0,957	0,959	0,959	
45	0,954	0,964	0,967	0,967	
58	0,958	0,966	0,968	0,968	
71	0,961	0,969	0,972	0,972	
Średnio	0,951	0,962	0,965	0,965	

BBB FLOWERS

		PSNR [dB]			
Widok	VSRS	MVS	MVS	MVS	
	V SKS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
19	22,52	24,25	25,02	25,12	
32	22,60	25,83	26,74	26,87	
45	22,68	25,31	26,15	26,26	
58	21,89	23,75	24,89	24,94	
71	23,42	25,61	26,60	26,69	
Średnio	22,62	24,95	25,88	25,97	

Tabela A99. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A100. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSPS	MVS	MVS	MVS	
	V3K5	(2 widoki)	(4 widoki)	(wszystkie widoki)	
19	0,799	0,843	0,849	0,850	
32	0,808	0,864	0,872	0,874	
45	0,824	0,878	0,885	0,884	
58	0,834	0,877	0,885	0,887	
71	0,857	0,886	0,897	0,898	
Średnio	0,824	0,870	0,877	0,878	

POZNAN_BLOCKS

	PSNR [dB]				
Widok	WEDE	MVS	MVS	MVS	
	v 3K3	(2 widoki)	(4 widoki)	(wszystkie widoki)	
1	22,65	22,98	22,99	22,98	
2	23,30	23,78	23,73	23,75	
3	24,61	25,08	25,02	25,02	
4	25,55	25,74	25,75	25,76	
5	24,68	25,08	25,07	25,08	
6	25,99	26,44	26,31	26,32	
7	25,66	26,57	26,60	26,61	
8	25,77	26,29	26,18	26,17	
Średnio	24,78	25,25	25,21	25,21	

Tabela A101. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A102. Porównanie jakości syntezowanych widoków dla VSRS i MVS

		SSIM				
Widok	VSRS	MVS	MVS	MVS		
	VSKS	(2 widoki)	(4 widoki)	(wszystkie widoki)		
1	0,718	0,736	0,735	0,734		
2	0,755	0,771	0,769	0,769		
3	0,762	0,773	0,769	0,770		
4	0,802	0,812	0,811	0,811		
5	0,805	0,817	0,814	0,814		
6	0,819	0,829	0,828	0,827		
7	0,807	0,826	0,821	0,822		
8	0,815	0,828	0,827	0,827		
Średnio	0,785	0,799	0,797	0,797		

POZNAN_BLOCKS2

	PSNR [dB]			
Widok	VEDE	MVS	MVS	MVS
	V SKS	(2 widoki)	(4 widoki)	(wszystkie widoki)
1	33,08	32,05	33,54	33,68
2	31,89	31,91	33,58	33,68
3	25,40	29,62	31,21	31,33
4	28,41	28,34	30,19	30,23
5	28,72	28,76	30,32	30,37
6	27,61	29,04	30,77	30,89
7	30,00	28,90	30,56	30,61
8	31,71	30,53	32,07	32,14
Średnio	29,60	29,89	31,53	31,62

Tabela A103. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A104. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSRS	MVS	MVS	MVS	
		(2 widoki)	(4 widoki)	(wszystkie widoki)	
1	0,884	0,887	0,891	0,891	
2	0,879	0,879	0,883	0,883	
3	0,872	0,886	0,889	0,890	
4	0,843	0,852	0,853	0,852	
5	0,832	0,832	0,832	0,833	
6	0,835	0,830	0,831	0,832	
7	0,835	0,834	0,835	0,836	
8	0,864	0,871	0,871	0,871	
Średnio	0,856	0,859	0,861	0,861	

13.1.1.1. POZNAN_FENCING2

	PSNR [dB]				
Widok	VEDE	MVS	MVS	MVS	
	v 3K3	(2 widoki)	(4 widoki)	(wszystkie widoki)	
1	30,03	30,51	30,69	30,67	
2	32,41	33,37	33,56	33,59	
3	31,79	32,17	32,20	32,21	
4	26,02	27,93	28,21	28,23	
5	25,60	27,63	27,80	27,77	
6	29,58	29,87	30,08	30,05	
7	30,01	30,42	30,65	30,61	
8	31,50	32,19	32,34	32,35	
Średnio	29,62	30,51	30,69	30,69	

Tabela A105. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A106. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM			
Widok	VSPS	MVS	MVS	MVS
	VSRS	(2 widoki)	(4 widoki)	(wszystkie widoki)
1	0,891	0,895	0,894	0,895
2	0,888	0,906	0,905	0,905
3	0,886	0,895	0,896	0,895
4	0,847	0,857	0,858	0,857
5	0,851	0,848	0,848	0,848
6	0,868	0,869	0,868	0,868
7	0,869	0,874	0,873	0,873
8	0,893	0,906	0,904	0,904
Średnio	0,874	0,881	0,881	0,880

POZNAN_SERVICE2

	PSNR [dB]			
Widok	VEDE	MVS	MVS	MVS
	V SKS	(2 widoki)	(4 widoki)	(wszystkie widoki)
1	18,76	19,50	19,80	19,77
2	24,63	24,32	24,97	24,94
3	27,61	27,61	28,27	28,23
4	25,43	26,82	27,00	26,97
5	25,58	26,55	26,64	26,64
6	25,17	25,01	25,53	25,55
7	24,44	25,09	25,38	25,40
8	25,81	25,85	26,43	26,38
Średnio	24,68	25,09	25,50	25,48

Tabela A107. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A108. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSRS	MVS	MVS	MVS	
		(2 widoki)	(4 widoki)	(wszystkie widoki)	
1	0,764	0,772	0,770	0,771	
2	0,804	0,821	0,820	0,819	
3	0,854	0,864	0,864	0,864	
4	0,806	0,820	0,819	0,819	
5	0,821	0,832	0,832	0,831	
6	0,803	0,807	0,804	0,804	
7	0,798	0,814	0,809	0,810	
8	0,810	0,816	0,813	0,812	
Średnio	0,808	0,818	0,816	0,816	

BALLET

	PSNR [dB]			
Widok	WEDE	MVS	MVS	MVS
	V SIXS	(2 widoki)	(4 widoki)	(wszystkie widoki)
1	25,85	26,59	26,52	26,52
2	26,61	27,22	27,31	27,29
3	30,71	31,28	31,38	31,35
4	31,32	32,64	32,72	32,74
5	31,40	32,69	32,59	32,61
6	29,39	30,11	30,05	30,03
Średnio	29,21	30,09	30,10	30,10

Tabela A109. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A110. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSPS	MVS	MVS	MVS	
	VSRS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
1	0,778	0,795	0,795	0,796	
2	0,780	0,794	0,795	0,795	
3	0,870	0,885	0,884	0,885	
4	0,863	0,885	0,886	0,885	
5	0,872	0,886	0,887	0,887	
6	0,859	0,869	0,869	0,869	
Średnio	0,837	0,852	0,853	0,853	

BREAKDANCERS

	PSNR [dB]			
Widok	VSPS	MVS	MVS	MVS
	VSKS	(2 widoki)	(4 widoki)	(wszystkie widoki)
1	28,54	29,06	29,02	29,03
2	30,80	31,48	31,50	31,49
3	30,84	31,58	31,61	31,60
4	31,41	32,06	32,07	32,07
5	31,21	33,68	33,67	33,66
6	30,93	31,89	31,94	31,93
Średnio	30,62	31,62	31,64	31,64

Tabela A111. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A112. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM			
Widok	VSRS	MVS	MVS	MVS
	VSKS	(2 widoki)	(4 widoki)	(wszystkie widoki)
1	0,797	0,814	0,815	0,815
2	0,820	0,839	0,839	0,839
3	0,837	0,852	0,852	0,853
4	0,847	0,864	0,864	0,864
5	0,840	0,863	0,863	0,864
6	0,825	0,841	0,841	0,841
Średnio	0,828	0,845	0,846	0,846

SOCCER ARC

	PSNR [dB]			
Widok	WEDE	MVS	MVS	MVS
	VSRS	(2 widoki)	(4 widoki)	(wszystkie widoki)
2	22,05	22,54	22,68	22,54
3	21,58	22,41	22,40	22,51
4	22,39	23,07	23,26	23,36
5	18,58	19,03	19,40	19,27
6	17,97	18,23	18,45	18,37
Średnio	20,51	21,06	21,24	21,21

Tabela A113. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A114. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSDS	MVS	MVS	MVS	
	v 5R5	(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	0,756	0,787	0,785	0,787	
3	0,782	0,824	0,823	0,824	
4	0,784	0,827	0,825	0,827	
5	0,647	0,678	0,676	0,678	
6	0,602	0,660	0,657	0,660	
Średnio	0,714	0,755	0,753	0,755	

SOCCER LINEAR

	PSNR [dB]				
Widok	Vede	MVS	MVS	MVS	
	V SIXS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	33,69	33,37	33,63	33,65	
3	33,85	33,56	33,85	33,85	
4	34,53	34,40	34,64	34,63	
5	35,23	35,26	35,45	35,47	
6	33,58	33,57	33,74	33,79	
Średnio	34,18	34,03	34,26	34,28	

Tabela A115. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A116. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSRS	MVS	MVS	MVS	
		(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	0,892	0,891	0,891	0,891	
3	0,895	0,911	0,911	0,911	
4	0,918	0,917	0,916	0,916	
5	0,921	0,923	0,923	0,923	
6	0,885	0,888	0,888	0,888	
Średnio	0,902	0,906	0,906	0,906	

POZNAN_CARPARK

	PSNR [dB]				
Widok	WEDE	MVS	MVS	MVS	
	V SKS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	35,28	35,34	35,43	35,43	
3	33,11	33,30	33,38	33,37	
4	33,89	33,86	34,00	33,99	
5	32,10	33,24	33,28	33,26	
6	33,14	33,26	33,36	33,34	
Średnio	33,50	33,80	33,89	33,89	

Tabela A117. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A118. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSRS	MVS	MVS	MVS	
		(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	0,939	0,941	0,941	0,941	
3	0,930	0,932	0,931	0,932	
4	0,930	0,932	0,932	0,932	
5	0,918	0,929	0,929	0,929	
6	0,925	0,928	0,928	0,928	
Średnio	0,929	0,933	0,932	0,932	

POZNAN_STREET

	PSNR [dB]				
Widok	ok vere	MVS	MVS	MVS	
	VSRS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	34,83	35,83	35,86	35,86	
3	34,61	34,57	34,58	34,58	
4	35,86	35,81	35,80	35,81	
5	36,19	36,27	36,34	36,33	
6	35,93	35,99	36,01	36,01	
Średnio	35,48	35,69	35,72	35,72	

Tabela A119. Porównanie jakości syntezowanych widoków dla VSRS i MVS

Tabela A120. Porównanie jakości syntezowanych widoków dla VSRS i MVS

	SSIM				
Widok	VSRS	MVS	MVS	MVS	
	V SIXS	(2 widoki)	(4 widoki)	(wszystkie widoki)	
2	0,930	0,940	0,940	0,940	
3	0,931	0,932	0,932	0,932	
4	0,936	0,936	0,937	0,937	
5	0,938	0,940	0,940	0,941	
6	0,934	0,938	0,937	0,937	
Średnio	0,934	0,937	0,937	0,937	

A7. OCENA JAKOŚCI SYNTEZY PODCZAS WIRTUALNEJ NAWIGACJI

Patrz: rozdział 10.2.3.

BBB BUTTERFLY

Tabela A121. Porównanie jakości synte:	zowanych widoków dla VSRS i MVS
(wszystkie widoki wirtualne syntezowano na	a podstawie widoków 06, 19, 32, 45, 58, 71 i 84)

		PSNR [dB]			
		MVS	MVS		
Widok	VSRS	(bez wstępnego	(ze wstępnym		
	V SIXS	zwiększania	zwiększaniem		
		rozdzielczości)	rozdzielczości)		
07	34,40	37,66	38,15		
08	35,20	37,33	37,63		
09	33,27	37,08	37,43		
10	36,24	36,49	37,14		
11	35,64	36,47	37,22		
12	35,40	36,76	37,13		
13	35,47	36,74	36,97		
14	34,71	36,60	36,97		
15	36,06	36,69	37,15		
16	36,58	37,01	37,54		
17	38,19	37,67	38,18		
18	37,20	38,44	38,90		
19	-	_	-		
20	36,04	37,92	38,59		
21	36,37	37,50	38,06		
22	35,43	36,92	37,41		
23	35,72	36,75	37,21		
24	34,73	36,64	37,14		
25	34,02	36,57	37,05		
26	35,76	36,51	36,96		
27	36,27	36,49	36,69		
28	36,03	36,39	36,89		
29	35,64	36,87	37,20		
30	35,75	37,23	38,00		

	PSNR [dB]			
-		MVS	MVS	
Widok	VODO	(bez wstępnego	(ze wstępnym	
	V 3K3	zwiększania	zwiększaniem	
		rozdzielczości)	rozdzielczości)	
31	35,62	38,11	38,89	
32	_	_	-	
33	32,57	38,66	39,16	
34	33,65	37,77	38,38	
35	34,36	37,41	37,87	
36	36,90	37,06	37,48	
37	35,58	36,92	37,21	
38	35,82	36,90	37,19	
39	36,27	36,93	36,95	
40	35,82	36,78	37,14	
41	35,92	36,92	37,49	
42	36,53	37,34	37,73	
43	36,45	37,77	38,65	
44	37,75	39,05	39,20	
45	-	-	-	
46	37,59	39,14	39,87	
47	36,67	38,59	38,94	
48	37,12	37,80	38,65	
49	36,78	37,46	37,93	
50	36,52	37,14	37,88	
51	35,81	37,36	37,75	
52	36,11	37,27	37,94	
53	35,59	37,32	37,80	
54	35,68	37,48	37,95	
55	34,43	37,77	38,40	
56	34,72	38,42	38,86	
57	34,40	38,96	39,54	
58	_	_	_	
59	36,35	38,86	39,45	

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

	PSNR [dB]			
		MVS	MVS	
Widok	WCDC	(bez wstępnego	(ze wstępnym	
	V SINS	zwiększania	zwiększaniem	
		rozdzielczości)	rozdzielczości)	
60	37,47	38,75	39,17	
61	37,48	38,30	38,71	
62	36,88	38,08	38,48	
63	36,88	38,18	38,36	
64	36,45	38,22	38,18	
65	36,31	37,94	38,19	
66	36,68	38,28	38,30	
67	36,89	38,40	38,67	
68	36,70	39,15	39,17	
69	37,56	39,48	39,78	
70	38,00	40,05	40,00	
71	_	_	_	
72	40,56	40,07	41,14	
73	40,46	40,03	40,62	
74	39,08	39,01	39,70	
75	38,33	38,55	39,20	
76	36,89	38,42	38,93	
77	37,15	38,59	38,60	
78	35,94	38,31	38,67	
79	36,57	38,30	38,34	
80	36,24	38,38	38,49	
81	32,70	38,67	38,49	
82	35,60	39,25	38,63	
83	34,64	39,14	38,98	
Średnio	36,15	37,83	38,23	

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

		SSIM	
_		MVS	MVS
Widok	VODO	(bez wstępnego	(ze wstępnym
	v 5R5	zwiększania	zwiększaniem
		rozdzielczości)	rozdzielczości)
07	0,971	0,985	0,986
08	0,977	0,984	0,984
09	0,966	0,983	0,984
10	0,980	0,981	0,982
11	0,978	0,981	0,982
12	0,978	0,982	0,982
13	0,978	0,983	0,983
14	0,974	0,983	0,984
15	0,983	0,984	0,985
16	0,985	0,985	0,987
17	0,988	0,987	0,988
18	0,986	0,990	0,990
19	_	_	_
20	0,983	0,988	0,989
21	0,984	0,987	0,988
22	0,981	0,985	0,986
23	0,981	0,983	0,984
24	0,980	0,983	0,984
25	0,978	0,982	0,983
26	0,981	0,982	0,983
27	0,981	0,982	0,983
28	0,980	0,982	0,983
29	0,980	0,983	0,985
30	0,980	0,985	0,986

Tabela A122. Porównanie jakości syntezowanych widoków dla VSRS i MVS
(wszystkie widoki wirtualne syntezowano na podstawie widoków 06, 19, 32, 45, 58, 71 i 84)

Widok	SSIM		
	Vobo	MVS	MVS
		(bez wstępnego	(ze wstępnym
	V 5K5	zwiększania	zwiększaniem
		rozdzielczości)	rozdzielczości)
31	0,977	0,986	0,988
32	_	_	_
33	0,968	0,986	0,987
34	0,972	0,985	0,986
35	0,975	0,984	0,985
36	0,982	0,983	0,983
37	0,979	0,982	0,983
38	0,980	0,983	0,983
39	0,981	0,983	0,983
40	0,981	0,983	0,984
41	0,982	0,984	0,985
42	0,984	0,986	0,987
43	0,983	0,987	0,989
44	0,987	0,989	0,990
45	_	-	_
46	0,986	0,989	0,991
47	0,983	0,989	0,989
48	0,984	0,987	0,988
49	0,983	0,985	0,986
50	0,981	0,984	0,986
51	0,979	0,984	0,985
52	0,980	0,983	0,985
53	0,979	0,983	0,985
54	0,981	0,984	0,985
55	0,974	0,985	0,986
56	0,974	0,986	0,987
57	0,971	0,987	0,988
58	_	_	_
59	0,982	0,987	0,988

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

	SSIM		
-		MVS	MVS
Widok	VCDC	(bez wstępnego	(ze wstępnym
	VSRS	zwiększania	zwiększaniem
		rozdzielczości)	rozdzielczości)
60	0,983	0,986	0,987
61	0,982	0,985	0,986
62	0,981	0,984	0,985
63	0,980	0,983	0,984
64	0,980	0,983	0,984
65	0,980	0,983	0,984
66	0,980	0,984	0,984
67	0,981	0,984	0,985
68	0,979	0,985	0,986
69	0,982	0,986	0,987
70	0,982	0,988	0,987
71	_	_	_
72	0,988	0,988	0,989
73	0,988	0,988	0,989
74	0,985	0,987	0,987
75	0,984	0,985	0,987
76	0,980	0,985	0,986
77	0,982	0,985	0,986
78	0,979	0,984	0,986
79	0,982	0,984	0,985
80	0,982	0,985	0,986
81	0,971	0,985	0,986
82	0,981	0,987	0,987
83	0,978	0,987	0,987
Średnio	0,980	0,985	0,986

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

BBB FLOWERS

	PSNR [dB]		
Widok		MVS	MVS
	VCDC	(bez wstępnego	(ze wstępnym
	v 5K5	zwiększania	zwiększaniem
		rozdzielczości)	rozdzielczości)
7	26,62	28,66	28,89
8	27,47	28,33	28,48
9	26,33	27,94	28,05
10	26,92	27,77	27,90
11	26,63	27,90	28,00
12	25,59	27,93	27,93
13	26,04	28,18	28,15
14	26,40	28,18	28,36
15	26,76	28,34	28,49
16	26,96	28,58	28,77
17	26,95	28,61	28,88
18	27,60	29,12	29,30
19	-	_	_
20	27,56	29,29	29,43
21	26,61	28,24	28,48
22	26,08	27,94	28,09
23	25,76	27,75	27,95
24	25,86	27,90	28,10
25	25,85	27,79	27,86
26	25,74	27,87	27,94
27	25,90	28,11	28,34
28	25,35	28,38	28,69
29	26,97	28,68	28,99
30	26,74	28,96	29,34

Tabela A123. Porównanie jakości syntezowanych widoków dla VSRS i MVS (wszystkie widoki wirtualne syntezowano na podstawie widoków 06, 19, 32, 45, 58, 71 i 84)

	PSNR [dB]			
Widok		MVS	MVS	
	VCDC	(bez wstępnego	(ze wstępnym	
	V 3K3	zwiększania	zwiększaniem	
		rozdzielczości)	rozdzielczości)	
31	28,90	29,84	30,08	
32	_	_	_	
33	27,26	30,51	30,60	
34	27,53	29,89	30,03	
35	26,56	29,39	29,60	
36	26,88	29,05	29,41	
37	26,33	28,78	28,82	
38	26,07	28,69	29,07	
39	26,26	28,85	29,04	
40	26,18	29,01	29,26	
41	26,41	29,35	29,61	
42	26,37	29,49	29,67	
43	27,56	29,65	29,87	
44	28,94	30,58	30,82	
45	-	_	-	
46	27,71	30,73	30,90	
47	27,36	30,09	30,38	
48	25,65	29,96	30,08	
49	26,00	29,40	29,76	
50	25,88	29,22	29,60	
51	25,47	29,19	29,53	
52	25,29	29,04	29,51	
53	25,36	29,02	29,48	
54	25,70	29,13	29,56	
55	25,85	29,22	29,71	
56	27,11	29,51	29,97	
57	28,08	30,14	30,60	
58	_	_	_	
59	27,31	31,01	31,24	

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

	PSNR [dB]		
-		MVS	MVS
Widok	VCDC	(bez wstępnego	(ze wstępnym
	V 5K5	zwiększania	zwiększaniem
		rozdzielczości)	rozdzielczości)
60	25,54	29,95	30,21
61	26,79	29,77	29,99
62	24,44	29,71	29,97
63	26,29	29,86	29,87
64	25,85	29,23	29,28
65	25,89	28,98	29,31
66	26,48	29,24	29,52
67	26,64	29,16	29,93
68	27,20	29,65	30,34
69	27,87	30,50	31,27
70	29,33	31,12	31,81
71	_	_	_
72	28,68	30,94	31,25
73	28,40	30,57	30,90
74	28,49	29,92	30,51
75	27,68	29,35	30,03
76	27,10	29,19	30,04
77	27,15	29,21	30,11
78	26,85	28,90	30,09
79	27,40	28,73	30,10
80	27,94	28,90	29,72
81	27,06	29,36	29,82
82	28,79	30,36	30,81
83	28,80	31,23	31,84
Średnio	26,80	29,21	29,55

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

	SSIM		
-		MVS	MVS
Widok	VODO	(bez wstępnego	(ze wstępnym
	v 5R5	zwiększania	zwiększaniem
		rozdzielczości)	rozdzielczości)
07	0,894	0,931	0,933
08	0,902	0,920	0,923
09	0,881	0,914	0,916
10	0,901	0,909	0,911
11	0,897	0,908	0,911
12	0,881	0,908	0,910
13	0,888	0,910	0,911
14	0,892	0,913	0,915
15	0,901	0,916	0,918
16	0,908	0,921	0,923
17	0,912	0,924	0,927
18	0,922	0,934	0,936
19	_	_	_
20	0,917	0,936	0,937
21	0,900	0,923	0,927
22	0,893	0,918	0,921
23	0,888	0,915	0,918
24	0,887	0,913	0,916
25	0,886	0,913	0,916
26	0,886	0,915	0,916
27	0,893	0,918	0,920
28	0,878	0,922	0,925
29	0,909	0,929	0,932
30	0,902	0,935	0,938

Tabela A124. Porównanie jakości syntezowanych widoków dla VSRS i MVS
(wszystkie widoki wirtualne syntezowano na podstawie widoków 06, 19, 32, 45, 58, 71 i 84)

Widok
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

	SSIM			
		MVS	MVS	
Widok	VCDC	(bez wstępnego	(ze wstępnym	
	V 5K5	zwiększania	zwiększaniem	
		rozdzielczości)	rozdzielczości)	
60	0,911	0,957	0,961	
61	0,931	0,952	0,956	
62	0,892	0,946	0,950	
63	0,924	0,943	0,947	
64	0,919	0,941	0,944	
65	0,921	0,940	0,944	
66	0,926	0,941	0,945	
67	0,927	0,944	0,949	
68	0,939	0,951	0,957	
69	0,944	0,960	0,964	
70	0,958	0,967	0,972	
71	_	_	_	
72	0,950	0,965	0,968	
73	0,947	0,959	0,962	
74	0,948	0,949	0,955	
75	0,936	0,942	0,948	
76	0,923	0,937	0,943	
77	0,922	0,933	0,939	
78	0,911	0,930	0,937	
79	0,923	0,930	0,937	
80	0,930	0,933	0,939	
81	0,903	0,940	0,945	
82	0,928	0,948	0,953	
83	0,920	0,955	0,961	
Średnio	0,915	0,937	0,940	

Porównanie jakości syntezowanych widoków dla VSRS i MVS (ciąg dalszy)

A8. SYNTEZA CZASU RZECZYWISTEGO

Patrz: rozdział 11.3.

BBB BUTTERFLY

Tabela A125. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
19	28,99	31,54	32,14
32	31,07	32,08	32,68
45	31,95	32,39	32,99
58	31,89	32,41	33,01
71	32,36	33,50	34,10
Średnio	31,25	32,39	32,98

Tabela A126. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
widok	VSRS	rtMVS	MVS
19	0,937	0,945	0,947
32	0,947	0,953	0,955
45	0,954	0,955	0,957
58	0,958	0,953	0,955
71	0,961	0,964	0,966
Średnio	0,951	0,954	0,956

BBB FLOWERS

Tabela A127. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
19	22,52	24,04	24,48
32	22,60	25,64	26,08
45	22,68	24,96	25,41
58	21,89	23,84	24,29
71	23,42	25,32	25,77
Średnio	22,62	24,76	25,21

Tabela A128. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widol	SSIM		
WIGOK	VSRS	rtMVS	MVS
19	0,799	0,829	0,836
32	0,808	0,853	0,860
45	0,824	0,862	0,869
58	0,834	0,861	0,868
71	0,857	0,872	0,879
Średnio	0,824	0,856	0,863

POZNAN_BLOCKS

•			,
Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
1	22,65	22,69	22,81
2	23,30	23,36	23,47
3	24,61	24,79	24,90
4	25,55	25,54	25,66
5	24,68	24,84	24,95
6	25,99	25,98	26,09
7	25,66	26,23	26,34
8	25,77	25,82	25,93
Średnio	24,78	24,91	25,02

Tabela A129. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A130. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
	VSRS	rtMVS	MVS
1	0,718	0,722	0,729
2	0,755	0,755	0,762
3	0,762	0,762	0,769
4	0,802	0,795	0,802
5	0,805	0,792	0,799
6	0,819	0,810	0,817
7	0,807	0,797	0,804
8	0,815	0,801	0,808
Średnio	0,785	0,779	0,786

POZNAN_BLOCKS2

Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
1	33,08	32,48	33,18
2	31,89	32,31	33,01
3	25,40	30,09	30,80
4	28,41	28,77	29,48
5	28,72	29,04	29,74
6	27,61	29,65	30,36
7	30,00	29,16	29,86
8	31,71	30,86	31,57
Średnio	29,60	30,30	31,00

Tabela A131. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A132. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
	VSRS	rtMVS	MVS
1	0,884	0,886	0,887
2	0,879	0,878	0,879
3	0,872	0,887	0,888
4	0,843	0,850	0,851
5	0,832	0,826	0,827
6	0,835	0,833	0,834
7	0,835	0,827	0,828
8	0,864	0,869	0,870
Średnio	0,856	0,857	0,858

POZNAN_FENCING2

Widala	PSNR [dB]		
WICOK	VSRS	rtMVS	MVS
1	30,03	29,55	29,97
2	32,41	32,58	33,01
3	31,79	31,10	31,53
4	26,02	27,23	27,66
5	25,60	26,75	27,17
6	29,58	29,10	29,52
7	30,01	29,74	30,17
8	31,50	31,36	31,79
Średnio	29,62	29,68	30,10

Tabela A133. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A134. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
WIGOK	VSRS	rtMVS	MVS
1	0,891	0,870	0,872
2	0,888	0,888	0,890
3	0,886	0,888	0,890
4	0,847	0,847	0,849
5	0,851	0,845	0,847
6	0,868	0,866	0,868
7	0,869	0,864	0,866
8	0,893	0,897	0,899
Średnio	0,874	0,871	0,873

POZNAN_SERVICE2

Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
1	18,76	19,10	19,47
2	24,63	24,28	24,64
3	27,61	27,47	27,84
4	25,43	26,37	26,74
5	25,58	26,00	26,37
6	25,17	24,81	25,18
7	24,44	24,69	25,06
8	25,81	25,69	26,05
Średnio	24,68	24,80	25,17

Tabela A135. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A136. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok -	SSIM		
	VSRS	rtMVS	MVS
1	0,764	0,778	0,783
2	0,804	0,811	0,816
3	0,854	0,855	0,860
4	0,806	0,799	0,804
5	0,821	0,811	0,816
6	0,803	0,773	0,778
7	0,798	0,800	0,805
8	0,810	0,795	0,800
Średnio	0,808	0,803	0,808

BALLET

			·
Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
1	25,85	25,83	26,23
2	26,61	26,41	26,81
3	30,71	30,53	30,93
4	31,32	31,89	32,28
5	31,40	31,80	32,20
6	29,39	29,09	29,49
Średnio	29,21	29,26	29,66

Tabela A137. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A138. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
	VSRS	rtMVS	MVS
1	0,778	0,781	0,788
2	0,780	0,783	0,790
3	0,870	0,871	0,878
4	0,863	0,863	0,870
5	0,872	0,867	0,874
6	0,859	0,842	0,849
Średnio	0,837	0,834	0,841

BREAKDANCERS

		1 0	,
Widok .	PSNR [dB]		
	VSRS	rtMVS	MVS
1	28,54	28,80	28,81
2	30,80	31,18	31,19
3	30,84	31,16	31,17
4	31,41	31,73	31,74
5	31,21	33,28	33,29
6	30,93	31,53	31,53
Średnio	30,62	31,28	31,29

Tabela A139. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A140. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
WICOK	VSRS	rtMVS	MVS
1	0,797	0,798	0,799
2	0,820	0,830	0,831
3	0,837	0,846	0,847
4	0,847	0,850	0,851
5	0,840	0,833	0,834
6	0,825	0,812	0,813
Średnio	0,828	0,828	0,829

SOCCER ARC

Widok -	PSNR [dB]		
	VSRS	rtMVS	MVS
2	22,05	22,07	22,25
3	21,58	21,87	22,06
4	22,39	22,65	22,83
5	18,58	18,92	19,10
6	17,97	17,96	18,14
Średnio	20,51	20,69	20,88

Tabela A141. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A142. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok -	SSIM		
	VSRS	rtMVS	MVS
2	0,756	0,753	0,766
3	0,782	0,793	0,806
4	0,784	0,792	0,805
5	0,647	0,648	0,661
6	0,602	0,617	0,630
Średnio	0,714	0,720	0,733

SOCCER LINEAR

Tabela A143. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok -	PSNR [dB]		
	VSRS	rtMVS	MVS
2	33,69	33,50	33,53
3	33,85	33,64	33,67
4	34,53	34,69	34,72
5	35,23	35,28	35,31
6	33,58	33,62	33,66
Średnio	34,18	34,14	34,18

Tabela A144. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok -	SSIM		
	VSRS	rtMVS	MVS
2	0,892	0,890	0,891
3	0,895	0,909	0,910
4	0,918	0,920	0,921
5	0,921	0,919	0,920
6	0,885	0,885	0,886
Średnio	0,902	0,905	0,906

POZNAN_CARPARK

wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)			
Widok	PSNR [dB]		
	VSRS	rtMVS	MVS
2	35,28	35,31	35,32
3	33,11	33,16	33,17
4	33,89	33,66	33,68
5	32,10	32,95	32,97
6	33,14	33,04	33,06
Średnio	33,50	33,62	33,64

Tabela A145. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Tabela A146. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM		
	VSRS	rtMVS	MVS
2	0,939	0,945	0,947
3	0,930	0,928	0,930
4	0,930	0,935	0,937
5	0,918	0,920	0,922
6	0,925	0,929	0,931
Średnio	0,929	0,931	0,933

POZNAN_STREET

Tabela A147. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	PSNR [dB]			
	VSRS	rtMVS	MVS	
2	34,83	35,57	35,57	
3	34,61	34,33	34,34	
4	35,86	35,52	35,53	
5	36,19	36,19	36,19	
6	35,93	35,79	35,80	
Średnio	35,48	35,48	35,49	

Tabela A148. Jakość syntezowanych widoków dla VSRS, rtMVS i MVS (w przypadku MVS do syntezy użyto wszystkich widoków rzeczywistych; nie zastosowano wstępnego zwiększania rozdzielczości)

Widok	SSIM			
	VSRS	rtMVS	MVS	
2	0,930	0,930	0,930	
3	0,931	0,941	0,941	
4	0,936	0,936	0,936	
5	0,938	0,937	0,937	
6	0,934	0,932	0,932	
Średnio	0,934	0,935	0,935	