

# Nowe techniki kompresji wizji dla rzeczywistości wirtualnej – MPEG Immersive Video

New video compression techniques for virtual reality – MPEG Immersive Video



Artykuł towarzyszy referatowi plenarnemu na temat najnowszej normy kodowania wizji – ISO/IEC 23090 część 12: MPEG Immersive Video (MIV). Zastosowaniem tej normy jest kompresja danych wizyjnych z wielu kamer do wykorzystania w rzeczywistości wirtualnej. Norma kodowania MIV jest niezależna od kodeków wideo, tj. obejmuje przetwarzanie wstępne i końcowe wokół istniejących kodeków, takich jak AVC, HEVC i VVC. Autorzy artykułu – wieloletni członkowie grupy eksperckiej ISO/IEC JTC1/SC29 WG11 MPEG, aktualnie WG04 MPEG Video Coding – wyjaśniają zasadę działania kodera i dekodera MIV, opisując najistotniejsze narzędzia umożliwiające efektywną kompresję wizji zarejestrowanej systemem wielokamerowym. Ponadto autorzy skupiają się na wymaganiach praktycznych systemów rzeczywistości wirtualnej, prezentując zalety i możliwości wykorzystania dwóch podstawowych profili MIV: profilu głównego (MIV Main) oraz profilu zakładającego estymację głębi po stronie odbiorczej (MIV Geometry Absent).  
**Słowa kluczowe:** wizja wszechogarniająca, kodowanie wizji, rzeczywistość wirtualna, przetwarzanie wizji wielowidokowej

The paper accompanies the plenary lecture which provides an overview of the latest video coding standard – ISO / IEC 23090 Part 12: MPEG Immersive Video (MIV). An application of this standard is the compression of video from multiple cameras for virtual reality applications. The MIV encoding standard is independent of the video codecs, i.e., it consists of pre-processing and post-processing merged with using existing codecs such as AVC, HEVC, and VVC. The authors of the paper – long-term experts of the ISO/IEC JTC1/SC29 WG11 MPEG, currently WG04 MPEG Video Coding – explain the principles of the MIV encoder and decoder, describing the most important tools enabling effective compression of video recorded with a multi-camera system. In addition, the authors focus on the practical requirements of virtual reality systems, presenting the advantages and possibilities of using two basic MIV profiles: the main profile (MIV Main) and the profile assuming estimation of depth on the decoder side (MIV Geometry Absent).

**Key word:** immersive video, video coding, virtual reality, multiview video processing

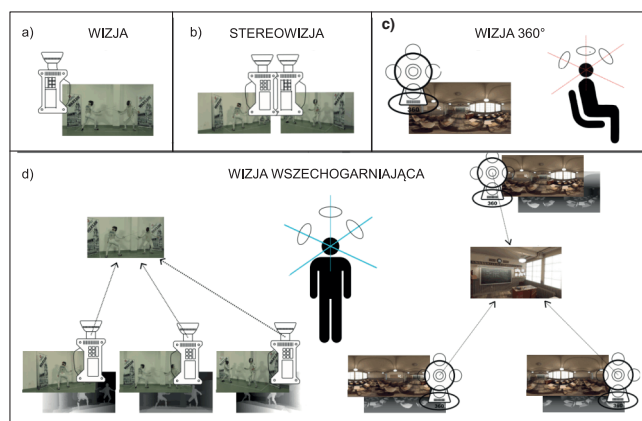
DOI: 10.15199/59.2022.4.4

## WIZJA WSZECHOGARNIAJĄCA

System wizji wszechogarniającej, zwanej immersyjną, jest systemem umożliwiającym użytkownikowi (wizdowi) wirtualne poruszanie się w scenie zarejestrowanej z użyciem wielu kamer. Taka nawigacja może być realizowana z wykorzystaniem wyświetlaczy nagłownych **HMD** (*Head-Mounted Displays*) [1], tradycyjnych monitorów z dołączonym urządzeniem sterującym, lub tabletów przez wykorzystanie ich dotykowych wyświetlaczy [2], [3].

W odróżnieniu od wizji 360°, gdzie widz może wyłącznie zmieniać kierunek obserwacji sceny (rys. 1b) i systemów stereoskopowych, które dodatkowo zapewniają wrażenie głębi (rys. 1c), w systemie wizji wszechogarniającej (rys. 1d) swoboda ruchów jest dużo większa, a widz może zarówno obracać się, jak i przemieszczać w scenie [4].

Pozornie może się wydawać, iż masowe wykorzystanie systemów wizji wszechogarniającej jest mrzonką lub – myśląc optymistycznie – kwestią dalekiej przyszłości, kojarzoną raczej z futurystycznymi filmami niż z dzisiejszymi transmisjami telewizyjnymi czy internetowymi. Myślenie takie jest jednak nieuzasadnione, a kolejne przykłady wykorzystania wirtualnej nawigacji w scenie regularnie pojawiają się w transmisjach zawodów sportowych z całego świata, np. transmisji koszykówki [5] czy też igrzysk olimpijskich, zarówno letnich [6], jak i zimowych [7].

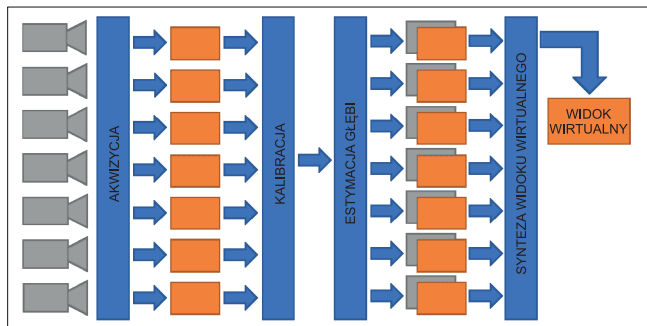


■ Rys. 1. Rodzaje systemów wizyjnych

Jeszcze częstsze są transmisje z uproszczonym modelem swobodnej nawigacji, w którym treść zarejestrowanej sceny jest zastępowana trójwymiarowym modelem boiska, pokazwanym przy rozstrzygnięciu spornych sytuacji z użyciem technologii *Goal Line* [8] czy tenisowej *HawkEye* [9].

Oczywiście obecne systemy nie wykorzystują pełni potencjału wizji wszechogarniającej, czyli umożliwienia użytkownikowi pełnej swobody ruchu w zarejestrowanej scenie. Niemniej jednak w wymienionych systemach wykorzystywane są te same

\* Instytut Telekomunikacji Multimedialnej, Politechnika Poznańska,  
 e-mail: adrian.dziembowski@put.poznan.pl, dawid.mieloch@put.poznan.pl



■ Rys. 2. Schemat przetwarzania wizji wszechogarniającej; kolorem niebieskim oznaczono kolejne kroki przetwarzania, pomarańczowym prostokątem – sekwencje wizyjne (widoki), a szarym – mapy głębi

techniki przetwarzania wizji wielowidokowej co w systemie wizji wszechogarniającej, a więc estymacja geometrii sceny i układu kamer oraz synteza (renderowanie) kolejnych obrazów na trajektorii położenia wirtualnej kamery. Kroki te zostały opisane w następnym rozdziale.

Jedyną – choć zasadniczą – różnicą jest wybór wspomnianej trajektorii, za którą obecnie odpowiada realizator transmisji, a nie końcowy użytkownik systemu, czyli widz. Różnica ta powoduje poważne konsekwencje pod względem szerokości pasma wymaganego do przesłania treści do widza. W przypadku wysyłania jednej wybranej trajektorii do wszystkich widzów transmitowany jest tradycyjny strumień wizyjny, a więc jedna sekwencja (na przykład FullHD zakodowana koderem HEVC [10]).

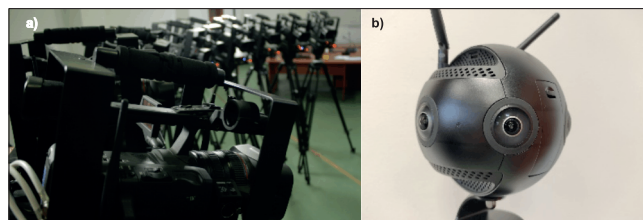
Przeniesienie możliwości wyboru trajektorii na użytkownika końcowego wiąże się natomiast z tym, iż każdy widz sam decyduje o tym, na co spogląda. Tym samym, aby umożliwić jednoczesną transmisję wizji wszechogarniającej do wielu użytkowników, nadawca nie może wybrać jednego, konkretnego strumienia wizyjnego. Zamiast tego konieczne jest wysłanie obrazów z wielu kamer.

W przypadku tak znaczącego zwiększenia ilości przesyłanych danych tradycyjne kodery wizyjne okazują się niewystarczająco efektywne, przez co stworzenie praktycznego systemu wizji wszechogarniającej bez wykorzystania dedykowanych algorytmów kompresji jest bardzo trudne, co przedstawiono w kolejnym rozdziale. Próbę rozwiązania tego problemu stanowi najnowsza norma kompresji – ISO/IEC 23090-12: MPEG Immersive Video [11], którą opisano w rozdziale *MPEG Immersive Video*.

## PRZETWARZANIE WIZJI WSZECHOGARNIAJĄCEJ

W celu zapewnienia użytkownikowi możliwości wirtualnej nawigacji pośród zarejestrowanej trójwymiarowej sceny należy uzyskać jej odpowiednią reprezentację. Najczęściej używana jest reprezentacja **MVD** (*Multiview Video plus Depth*) [12], w której zawartość oraz geometria sceny są przechowywane w postaci wielu widoków zarejestrowanych z pomocą kamer oraz tak zwanych map głębi (opisanych w rozdziale *Estymacja głębi*). Inne spotykane reprezentacje oparte są na chmurach trójwymiarowych punktów [13], które są częściej wykorzystywane dla danych uzyskanych za pomocą sprzętowych metod akwizycji geometrii sceny (np. systemów LiDAR [14]).

Odpowiednia reprezentacja sceny, wraz z parametrami i pozycjami użytych kamer, wykorzystywana jest do stworzenia dla użytkownika wybranego widoku w procesie syntezy widoku wirtualnego. Uproszczony schemat przetwarzania przedstawio-



■ Rys. 3. System wielokamerowy wyposażony w tradycyjne, perspektywiczne kamery (a); pojedyncza kamera wszechkierunkowa (b)

no na rys. 2, a dalsze części tego rozdziału opisują poszczególne etapy przetwarzania wraz z przykładowymi metodami używanymi w rzeczywistych systemach wizji wszechogarniającej.

### Akwizycja

W systemach wizji wszechogarniającej sposób rozmieszczenia kamer oraz ich typ nie są ściśle ograniczone i w głównej mierze zależą od rozmiaru rejestrowanej sceny i końcowego przeznaczenia systemu. W przypadku użycia typowych, standardowych kamer perspektywicznych (rys. 3a) ich liczba może sięgać setek [2], ale w praktycznych systemach jest takich kamer kilka/kilkanaście w zależności od użytego rozstawienia [3].

Używając kamer wszechkierunkowych [1] (rys. 3b), można zredukować liczbę urządzeń, gdyż każde z nich rejestruje dużo większy obszar sceny niż kamera perspektywiczna. Z drugiej strony ich użycie jest często niemożliwe w przypadku systemu realizowanego na potrzeby np. transmisji sportowych, w których kamery nie mogą być umieszczone wewnątrz rejestrowanej sceny, lecz dookoła niej.

Sekwencje używane w wizji wszechogarniającej mogą być również oparte na wykorzystaniu grafiki komputerowej, zarówno przez generowanie odpowiednich scen w całości, jak przez łączenie takich scen z rzeczywistymi obiektami zarejestrowanymi przez kamery (np. przez umieszczenie naturalnych postaci w komputerowej scenarii).

### Kalibracja

W celu uzyskania parametrów kamer, konieczna jest ich kalibracja. Kalibracja dla parametrów wewnętrznych kamer, która m.in. obejmuje estymację ich ogniskowej czy zniekształceń wprowadzanych przez soczewki kamer, jest łatwa do wykonania przy użyciu zdefiniowanego wzorca, np. klasycznej szachownicy. Zazwyczaj dla każdej kamery wymagane jest zarejestrowanie dziesięciu różnych widoków wzorca. Wiele metod dostępnych w bibliotekach *open source* zapewnia wystarczającą dokładność do zastosowań wizji wszechogarniającej (np. OpenCV [15]).

Kalibracja parametrów zewnętrznych kamer, tj. opisujących położenie kamer względem rejestrowanej sceny, powinna być powtarzana każdorazowo po zmianie pozycji kamer. Zazwyczaj jest wykonywana przez optymalizację funkcji opisującej zależności między punktami charakterystycznymi zarejestrowanymi jednocześnie przez kamery systemu (minimalizacja odległości od punktu do linii epipolarnej [16]). Automatyczne wyszukiwanie takich punktów odniesienia jest preferowane w praktycznych systemach wielokamerowych i jest uzyskiwane przez automatyczną ekstrakcję cech lub używanie pewnych własności obiektu kalibracyjnego, np. najjaśniejszego punktu w scenie w przypadku użycia źródła światła.

W przypadku scen zarejestrowanych za pomocą rzeczywistych kamer dobrą praktyką jest przeprowadzenie również tzw. korekcji koloru [17], która ma zminimalizować różnice w charakterystyce barwnej, występujące pomiędzy kamerami systemu, jak i zmiany charakterystyki w czasie (wynikające np. z użycia automatycznego balansu bieli). Jednoczesne zmniejszenie międzywidokowej i czasowej fluktuacji charakterystyk barwnych ma pozytywny wpływ na jakość estymowanych map głębi i wydajność kompresji takich sekwencji [18].

### Estymacja głębi

Mapy głębi, najczęściej reprezentowane jako obrazy w skali szarości, przedstawiają odległość punktów zarejestrowanej sceny od płaszczyzny opisanej na przetworniku kamery (rys. 4). Mapy głębi najczęściej estymowane są algorytmicznie na podstawie widoków zarejestrowanej sceny. Głębina punktu jest wyliczana przez poszukiwanie punktu odpowiadającego, czyli punktu, który reprezentuje tę samą część sceny w innym widoku [16].

Biorąc pod uwagę dużą złożoność obliczeniową tego etapu przetwarzania sekwencji wielowidokowych, mapy głębi często są również uzyskiwane za pomocą tzw. kamer głębi [14] lub za pomocą metod opartych na zastosowaniu głębokiego uczenia [19]. W obu tych przypadkach często wymagane jest zastosowanie dodatkowego przetwarzania końcowego, mającego na celu międzywidokowe uwspólnienie map głębi, które są estymowane niezależnie dla każdego z widoków [20].

### Synteza widoków wirtualnych

Synteza widoków wirtualnych stanowi ostatni etap przetwarzania danych w systemie wizji wszechogarniającej. W syntezie wielowidokowej [21], [22] zmarginalizowany jest wpływ przysłonięć obiektów. Kiedy scena rejestrowana jest przez większą liczbę kamer, obszarów całkowicie przysłoniętych we wszystkich widokach rzeczywistych jest niewiele. Metoda syntezy powinna być jednak dostosowana do systemu, w którym będzie wykorzystywana, np. przez umożliwienie syntezy z obrazów wszechkierunkowych [23], [24] czy syntezę dokonywaną w czasie rzeczywistym [25], [26], niezbędną w przypadku użycia wyświetlaczy nagłownych do obserwowania sceny.

### KOMPRESJA WIZJI WSZECHOGARNIAJĄCEJ

Powszechnie wiadomo, że niezależna kompresja wielu widoków i głębi skutkuje wysokimi przepływnościami. Co więcej, nowym krytycznym ograniczeniem w wizji wszechogarniającej jest tak zwana prędkość próbkowa (*pixel rate*): odzwierciedla całkowitą liczbę próbek obrazu, które mają być zdekodowane na sekundę w celu zainicjowania syntezy nowego wirtualnego widoku. Jest to zwykle uwarunkowane praktycznymi zasobami dekoderów sprzętowych. Na przykład, dla normy HEVC, profil Main 10 na poziomie 5.2 [10], który jest kompatybilny z obecnymi sprzętowymi dekoderni (np. [27], [28]), zakłada, że dla sekwencji o rozdzielczości 4096×2048 dla 30 klatek na sekundę liczba wystąpień dekodera nie może być większa niż cztery. W przypadku wizji wszechogarniającej oznacza to, że skompresowany strumień nie może zawierać więcej niż dwa widoki i odpowiadające im mapy głębi. Co więcej, podczas gdy nowoczesne uniwersalne kodery wideo zapewniają dobrą wydajność kodowania typowych treści zarejestrowanych przez kamery, kodowanie mapy głębi jest zdecydowanie bardziej wydajne,

gdy stosowane są dedykowane rozwiązania. Z tego powodu używanie tego samego kodera do obu celów jest niewłaściwe.

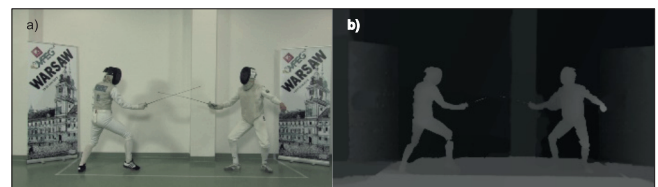
Jak opisywano wcześniej, wielowidokowa sekwencja jest rejestrowana przez kilka kamer (np. wszechkierunkowych), w których znacząca część sceny jest widoczna we wszystkich z nich. Nadmiarowość wynikająca z przestrzennego nakładania się widoków wejściowych może być wykorzystana do zmniejszenia rozmiaru danych wymaganych do pełnego odwzorowania całej trójwymiarowej sceny. Pierwsze kodery wizyjne, mające w wydajny sposób kompresować obrazy wielowidokowe przez identyfikację takich wspólnych treści, czyli MVC [29], MV-HEVC i 3D-HEVC [30], które są rozszerzeniami szeroko stosowanych norm kompresji typowych jednowidokowych sekwencji, nie są niestety dopasowane do charakterystyki wizji wszechogarniającej. Zarówno MVC, jak MV-HEVC, nie zapewniają usprawnień dla kodowania głębi (która może stanowić ponad 50% strumienia bitowego). 3D-HEVC, mimo że jego implementacje są nadal aktywnie ulepszone przez niektórych badaczy, nie jest kompatybilny z nieliniowymi układami kamer i kamerami wszechkierunkowymi, znacznie ograniczając jego użyteczność. Z tego powodu nie zaplanowano takiego typu rozszerzeń dla najnowszej normy *Versatile Video Coding (VVC)*, lecz zdecydowano się na nowe podejście do kodowania wizji wszechogarniającej, czyli *MPEG Immersive Video (MIV)*.

### MPEG IMMERSIVE VIDEO

Obecna, najnowocześniejsza technologia kompresji dla wizji wszechogarniającej została opracowana przez grupę MPEG ISO/IEC pod nazwą *MPEG Immersive Video* [31]. Główną ideą w tej normie jest założenie, że pewna mała liczba widoków bazowych, gromadzących większość informacji o scenie, powinna być zakodowana w całości, a informacje uzupełniające (widoczne z pozostałych widoków) mogą być przekazywane w formie małych fragmentów („łatek”), tworzących „atlas”, bądź pominięte, zwiększając liczbę widoków bazowych.

Z formalnego punktu widzenia, norma ISO/IEC 23090:12 MIV [11] stanowi rozszerzenie normy ISO/IEC 23090:5 V3C (*Visual Volumetric Video-based Coding*) [13] standaryzującej kodowanie scen trójwymiarowych reprezentowanych w postaci chmur punktów (*point clouds*). Kodowanie MIV wykorzystuje składnię i semantykę normy V3C, dodając techniki i narzędzia niezbędne do poprawnego kodowania i dekodowania treści w formacie MVD [12].

W odróżnieniu od wcześniejszych koderów wizji wielowidokowej (MVC, MV-HEVC i 3D-HEVC), które definiowały cały proces kodowania, począwszy od wczytania widoków wejściowych aż do tworzenia jednego wspólnego strumienia bitowego, koder MIV stanowi swoiste przetwarzanie wstępne wizji wielowidokowej, usuwając redundancję międzywidokową i zmieniając sposób reprezentacji danych wejściowych. Wyjście z kodera MIV stanowią atlasy – sekwencje wizyjne zawierające informacje



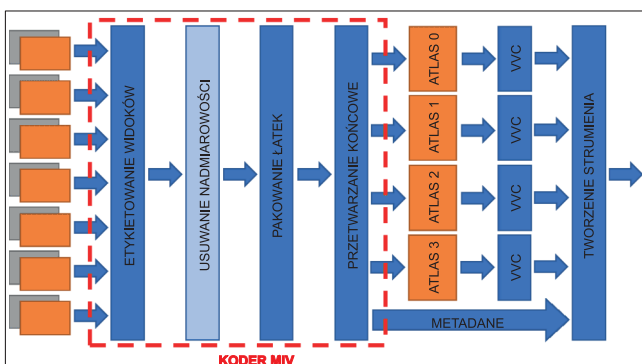
■ Rys. 4. Widok zarejestrowany przez kamerę (a) i odpowiadająca mu mapa głębi (b), sekwencja *Fencing*

z wielu widoków wejściowych. Atlasy te są następnie kodowane przy użyciu dowolnego kodera wizyjnego, czyniąc MIV techniką całkowicie niezależną od używanej techniki kompresji (*codec-agnostic*) [32]. Tym samym, wraz z MIV, można użyć efektywnej, choć czasochłonnej kompresji, z użyciem najnowszego kodera VVC [33], jak również mniej efektywnych, ale znacznie szybszych koderów AVC [34] czy HEVC [10] z dowolnie ustawionymi parametrami [35] czy włączonymi dodatkowymi narzędziami [36]. Pełna dowolność w odniesieniu do używanego kodera wizyjnego stanowi niepodważalną zaletę MIV.

Norma MPEG Immersive Video definiuje 3 profile: główny (MIV Main), rozszerzony (MIV Extended) oraz profil z estymacją głębi po stronie dekodera (MIV Geometry Absent, GA). Poszczególne profile odpowiadają różnym wymaganiom rynku i potrzebom użytkowników. W niniejszym artykule skupiono się na dwóch z nich – MIV Main oraz MIV GA.

## KODOWANIE MIV

Jak wspomniano, koder MIV stanowi przetwarzanie wstępne sekwencji wielowidokowej, umieszczając informację z widoków wejściowych w czterech atlasach. Następnie każdy z atlasów jest niezależnie kodowany z użyciem typowego kodera wizyjnego (np. VVC [33]), a skompresowane strumienie wizyjne, wraz ze strumieniem metadanych, są multipleksowane w celu stworzenia jednego strumienia bitowego transmitowanego do użytkownika. Schemat kodowania wizji z użyciem kodera MIV przedstawiono na rys. 5.



■ Rys. 5. Uproszczony schemat kodowania MIV, czerwoną ramką zaznaczono koder MIV, kolorem niebieskim oznaczono kolejne kroki kodowania (jasnoniebieskim – kroki opcjonalne), pomarańczowym – sekwencje wizyjne przed i po kodowaniu koderem MIV

Wyodrębniono cztery podstawowe etapy przetwarzania wizji w koderze MIV (czerwona ramka): etykietowanie widoków, usuwanie nadmiarowości, pakowanie łątek do atlasów i przetwarzanie końcowe. Kroki te zostały opisane w kolejnych podrozdziałach.

Szczegółowy opis kodera MIV można znaleźć w artykule [31] oraz dokumencie opisującym działanie modelu testowego MIV [37].

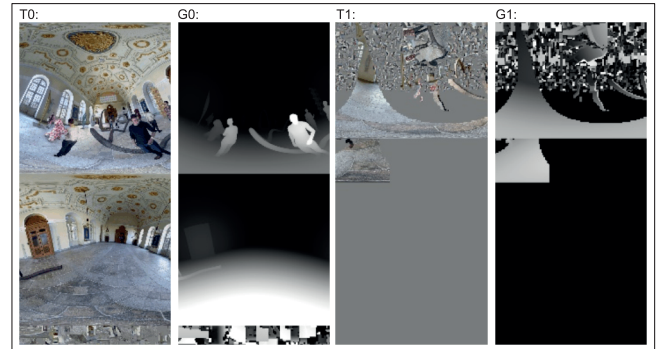
### MIV Main

Profil główny normy MIV (MIV Main) opisuje podstawowy sposób kodowania wizji wszechogarniającej, przeznaczony dla prostych, praktycznych systemów swobodnej nawigacji. W takich systemach kluczowe jest szybkie dekodowanie treści z użyciem prostych i tanich urządzeń końcowych, np. telefonów komórkowych czy tabletów.

Koder MIV, działający w trybie MIV Main, realizuje wszystkie cztery główne kroki przetwarzania wizji wielowidokowej, generując dwa atlasy tekstury i dwa atlasy głębi (rys. 6). Atlasy te zawierają informację ze wszystkich widoków wejściowych.

### Etykietowanie widoków

W pierwszym kroku kodowania koder MIV decyduje, które z widoków wejściowych niosą najwięcej istotnych informacji. Widoki te są określane widokami bazowymi i będą umieszco-



■ Rys. 6. Cztery atlasy wygenerowane przez koder MIV działający w trybie MIV Main; sekwencja *Museum*

ne w atlasach w całości (2 widoki w atlasach T0 i G0 na rys. 6). Istotna (nienadmiarowa) informacja z pozostałych widoków wejściowych (określanych widokami dodatkowymi) również jest umieszczana w atlasach, aczkolwiek widoki te są wcześniej przetwarzane w celu usunięcia redundancji międzywidokowej.

Liczba widoków bazowych zależy od rozdzielczości sekwencji. W przypadku sekwencji wysokiej rozdzielczości (4K) wysłanie większej liczby widoków bazowych niż jeden spowodowałoby, iż w atlasach zabrakłoby miejsca na informację z pozostałych widoków. Dla sekwencji o rozdzielczości mniejszej (np. FullHD) widoków bazowych jest więcej.

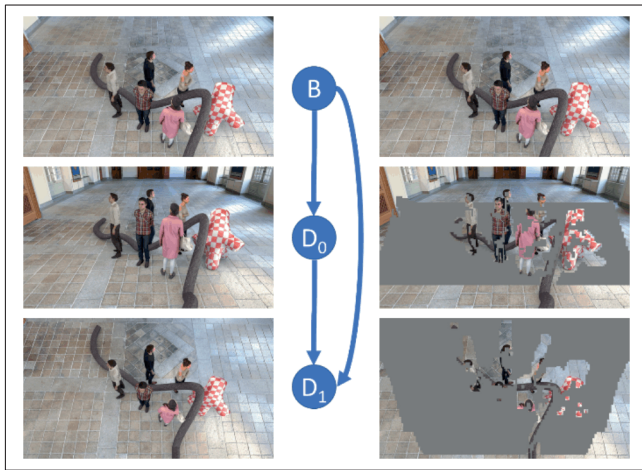
Wybór widoków bazowych dokonywany jest w sposób automatyczny na podstawie analizy rozmieszczenia kamer w systemie. Jeżeli jest tylko jeden widok bazowy, wybierany jest widok z najbardziej centralnej kamery systemu. W przypadku większej liczby ( $k$ ) widoków bazowych, wybierane są widoki z  $k$  kamer najbardziej oddalonych od siebie. Widoki bazowe wybierane są przy użyciu algorytmu PAM (*Partitioning Around Medoids*) [38].

### Usuwanie nadmiarowości

W celu usunięcia międzywidokowej nadmiarowości widoków dodatkowych, stosuje się przerzutowanie punktów (pikseli) pomiędzy poszczególnymi widokami.

W pierwszym kroku wszystkie punkty widoku bazowego (widoków bazowych, węzeł B na rys. 7) są rzutowane do każdego widoku dodatkowego. W przypadku, gdy kolor i głębia punktu ( $x, y$ ) w widoku dodatkowym są takie same jak kolor i głębia przerzutowane do punktu ( $x, y$ ) z widoku bazowego, punkt ten jest uznawany za nadmiarowy i usuwany (szare obszary na rys. 7) [39].

Następnie wybierany jest ten spośród widoków dodatkowych (węzeł  $D_0$  na rys. 7), w którym usunięto najmniej nadmiarowych punktów, a cały algorytm jest iteracyjnie powtarzany przez przerzutowanie nienadmiarowych punktów z tego widoku do pozostałych widoków dodatkowych. Algorytm kończy się w momencie, gdy usunięto nadmiarowe punkty ze wszystkich widoków wejściowych.



■ Rys. 7. Proces usuwania nadmiarowości międzywidokowej. Po lewej – widoki wejściowe: B (bazowy),  $D_0$  i  $D_1$  (widoki dodatkowe), po prawej – widoki po usunięciu informacji nadmiarowej (szare obszary). Strzałkami oznaczono kierunek rzutowania informacji pomiędzy widokami

### Pakowanie łątek do atlasów

Po usunięciu nadmiarowości widoki dodatkowe zawierają wyłącznie informację nienadmiarową. Informacja ta, wraz z pełnymi widokami bazowymi, jest umieszczana w atlasach w postaci mozaiki łątek, gdzie łątka oznacza prostokątny obszar widoku wejściowego obejmujący wzajemnie połączone nieusunięte punkty. W przypadku widoku bazowego łątka obejmuje cały obszar widoku.

Poszczególne łątki są sekwencyjnie pakowane do atlasów w kolejności malejącej powierzchni, z użyciem algorytmu MaxRect [40].

Oczywiście, w celu zapewnienia dekodowalności, algorytm pakowania musi być odwracalny, a więc informacja o rozmiarze i oryginalnym położeniu każdej łątki w widoku wejściowym musi zostać przesłana. W koderze MIV informacja ta jest umieszczana w strumieniu metadanych (rys. 5).

### Przetwarzanie końcowe

Atlasy stworzone w procesie pakowania zawierają znacznie mniej informacji niż widoki wejściowe. Niemniej jednak charakterystyka atlasu (wiele małych obiektów, krawędzi i wysokich częstotliwości) czyni je treścią bardzo trudną do efektywnego zakodowania z użyciem tradycyjnych koderów wizyjnych.

Dlatego też przed zakodowaniem atlasy są dodatkowo przetwarzane. W przypadku głębi rozdzielczość atlasów jest dwukrotnie zmniejszana w obu kierunkach, a zakres dynamiczny głębi jest normalizowany. Poszczególne łątki w atlasach tekstury są natomiast modyfikowane przez usunięcie składowej stałej lamy i obu chrom, zmniejszając amplitudę i liczbę krawędzi pomiędzy łątkami.

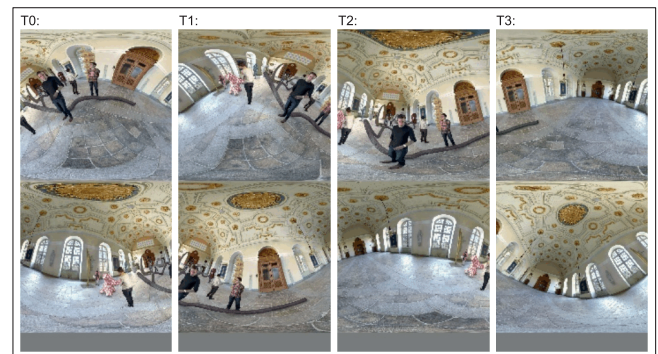
Ponadto zarówno w głębi, jak i teksturze, łątki nie zawierają wyłącznie obszarów nieusuniętych (o nieregularnych krawędziach), ale również fragmenty ich otoczenia [41], dostosowując się do działania koderu wizyjnego, który operuje na strukturze blokowej jednostek CU [10].

### MIV Geometry Absent

Inne podejście do kodowania wizji wszechogarniającej polega na zastąpieniu transmisji map głębi ich estymacją ze zdeko-

dowanych widoków w dekoderze lub w chmurze obliczeniowej [42]. Takie rozwiązanie umożliwia uwzględnienie w strumieniu bitów dwukrotnie większej liczby atlasów tekstury niż w typowej kompresji widoków i map głębi za pomocą MIV Main (por. rys. 6 i 8), przy tej samej prędkości próbkowej.

Profil MIV GA umożliwia strumieniom bitowym MIV nie zawieranie map głębi, zachowując tylko informacje wymagane do ich rekonstrukcji. Szczegóły dotyczące wszystkich rozwiązań znajdujących zastosowanie w tym profilu można znaleźć w [43].



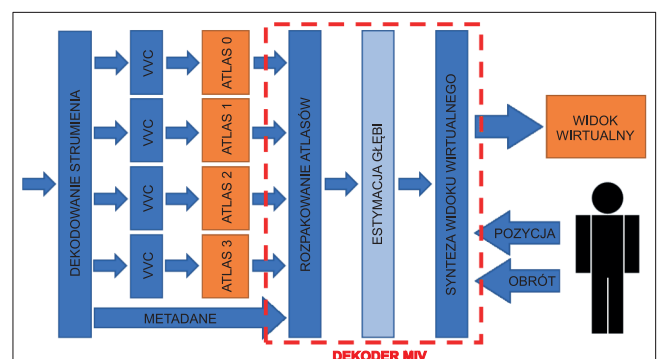
■ Rys. 8. Cztery atlasy wygenerowane przez koder MIV działający w trybie MIV GA; sekwencja Museum

### Etykietowanie i pakowanie widoków

W przypadku estymacji głębi po stronie dekodera nie jest przeprowadzane usuwanie nadmiarowości, gdyż jest ono dokonywane na podstawie map głębi widoków wejściowych, które nie muszą być dostępne w koderze. Zatem podczas etykietowania widoków wybierane są tylko widoki podstawowe, jednak w większej liczbie niż w przypadku podstawowego profilu, gdyż nie wysyłając głębi, możliwe jest umieszczenie dodatkowych pełnych widoków w atlasach bez zmiany prędkości próbkowej. W związku z używaniem jedynie pełnych widoków, ich pakowanie sprowadza się do umieszczenia ich kolejno w atlasie, znacząco upraszczając ten krok i zmniejszając złożoność koderu.

### Przetwarzanie końcowe

Jedną z wad profilu MIV GA jest brak możliwości skorzystania z wysokiej jakości map głębi, które mogą znajdować się po stronie koderu. Te mapy mogą być dostarczane wraz sekwencjami wygenerowanymi komputerowo lub estymowane



■ Rys. 9. Uproszczony schemat dekodowania MIV, czerwoną ramką zaznaczono dekoder MIV, kolorem niebieskim oznaczono kolejne kroki dekodowania (jasnoniebieskim – kroki opcjonalne), pomarańczowym – sekwencje wizyjne przed i po dekodowaniu dekoderem MIV

za pomocą ręcznych lub automatycznych metod poprawy ich jakości w przypadku treści naturalnych.

Aby możliwe było wykorzystanie tych wartościowych danych, są one przesyłane w strukturze Geometry Assistance SEI, umożliwiającej transmisję pewnych informacji pobocznych, które pomagają estymatorowi głębi po stronie dekodera w uzyskaniu map głębi o wyższej jakości przy jednoczesnym zmniejszeniu złożoności estymacji [44]. Jedną z przykładowych przesyłanych informacji jest flaga wskazująca dla poszczególnych bloków mapy głębi, czy można użyć dla danego obszaru głębię z poprzedniej ramki w związku z brakiem ruchomych obiektów.

### Kodowanie atlasów

Jak opisano wcześniej, do zakodowania atlasów stworzonych przez koder MIV można wykorzystać dowolną technikę kodowania wizji. W związku z tym każdy atlas kodowany jest niezależnie, umożliwiając traktowanie każdego z nich w taki sam sposób jak tradycyjnej sekwencji wizyjnej. Na koniec powstałe w koderze MIV metadane są multipleksowane ze strumieniami wideo, tworząc jeden wspólny strumień bitowy.

## DEKODOWANIE MIV

Tak jak koder MIV stanowi przetwarzanie wstępne sekwencji wielowidokowej, tak dekodery MIV jest przetwarzaniem końcowym po zdekodowaniu przesłanych strumieni wizyjnych (rys. 9).

Po zdemultipleksowaniu strumienia otrzymywane są 4 strumienie wizyjne oraz strumień metadanych. Każdy ze strumieni wizyjnych jest dekodowany z użyciem typowego dekodera wizyjnego, a zdekodowane sekwencje (atlasy) wraz z metadanymi są następnie przetwarzane przez dekodery MIV w celu wygenerowania widoku oglądanego przez końcowego użytkownika systemu wizji wszechogarniającej.

### MIV Main

Jak podkreślono wcześniej, podstawowym zastosowaniem profilu głównego są proste, praktyczne systemy, w których kluczowa jest szybkość dekodowania. W związku z tym dekodery MIV, działający w trybie MIV Main (rys. 9) jest uproszczony i składa się z dwóch podstawowych etapów: rozpakowywania atlasów i syntezy widoków wirtualnych.

#### Rozpakowanie atlasów

Pierwszym etapem przetwarzania sekwencji wielowidokowej w dekoderyze MIV jest odtworzenie widoków wejściowych, w których umieszczane są wszystkie łuki przesłane w ramach atlasów. Oczywiście nie wszystkie widoki są odtwarzane w całości, ponieważ część powierzchni widoków dodatkowych została beżpowrotnie usunięta w koderze, gdyż była nadmiarowa.

#### Synteza widoku wirtualnego

W drugim i ostatnim etapie dekodowania dokonywana jest synteza widoku wirtualnego prezentowanego widzowi. Wykorzystywane są do niej odtworzone widoki wejściowe, parametry kamer przesłane w strumieniu metadanych oraz informacja o położeniu i kierunku obserwacji sceny przez użytkownika. Sam proces syntezy nie jest zdefiniowany w normie MIV, a więc do wygenerowania widoku wirtualnego może zostać użyty – w zależności od potrzeb – dowolny syntezer. W szczególności, w praktycznym systemie wizji wszechogarniającej, w którym

dekoder powinien działać w czasie rzeczywistym i generować widok wirtualny zaraz po aktualizacji położenia widza, powinien zostać użyty syntezer czasu rzeczywistego, np. [26], [24].

W oprogramowaniu modelowym MIV [37] zastosowany został algorytm VWS [25], działający wolniej, lecz umożliwiający uzyskanie wirtualnych widoków o wysokiej jakości.

## Dekoder MIV Geometry Absent

W związku z koniecznością przeprowadzenia czasochłonnego procesu estymacji map głębi, dekodery w profilu MIV GA jest zdecydowanie bardziej wymagający pod względem złożoności obliczeniowej niż dekodery dla MIV Main [42]. Chociaż złożoność obliczeniowa estymacji głębi może być postrzegana jako wada używania tego profilu, możliwe jest znaczne ograniczenie jej przez estymację tylko dla części widoków, które są niezbędne do syntezy żądanego widoku wirtualnego. Bez dodatkowego kanału sprzężenia zwrotnego jest to niemożliwe w MIV Main, dla którego estymacja głębi po stronie kodera dla wszystkich widoków jest obciążeniem dla takich aplikacji, jak strumieniowanie wydarzeń na żywo.

Obecnie w badaniach dotyczących normy MIV, do celów estymacji głębi (zarówno po stronie kodera, jak i dekodera), wykorzystywane jest oprogramowanie **IVDE** (*Immersive Video Depth Estimation*) [45], które stanowi rozwinięcie metody estymacji głębi opartej na segmentacji obrazu [46]. Metoda ta jest rozwijana równolegle z modelem testowym MIV i zapewnia ulepszenia takie, jak zwiększenie jakości map głębi dla skompresowanych widoków wejściowych [47], zwiększające jej użyteczność w badaniach nad MIV GA.

## BADANIA EKSPERYMENTALNE

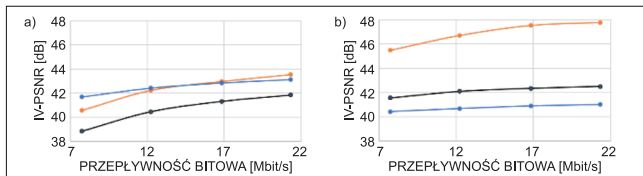
Przedstawiono tu porównanie obu profili normy MIV, zestawiając je z istniejącym podejściem, w którym widoki wejściowe i towarzyszące im głębie kodowane są w sposób niezależny, bez jakiegokolwiek redukcji redundancji międzywidokowej (*multi-view simulcast*).

### Metodologia badań

Eksperymenty zostały przeprowadzone, zgodnie z warunkami ustalonymi przez grupę ISO/IEC MPEG Video Coding (VC) [48], na zbiorze testowym zawierającym 15 zróżnicowanych sekwencji testowych, wliczając w to sekwencje naturalne zarejestrowane rzeczywistymi systemami wielokamerowymi i wygenerowane komputerowo, zarówno tradycyjne (perspektywiczne), jak i wszechkierunkowe (360°).

Do eksperymentów użyto oprogramowania modelowego MIV [37], a stworzone atlasy kodowano przy użyciu szybkiej implementacji kodera VVC [33]: VVenC [49]. W przypadku profilu MIV GA dodatkowo użyto oprogramowania IVDE [45] do estymacji głębi po stronie dekodera.

Każdą sekwencję testową zakodowano czterokrotnie, z różnymi wartościami parametru kwantyzacji QP [10], pokazując efektywność kompresji kodowania MIV dla różnych przepływności bitowych. Jakość syntezowanych widoków wirtualnych mierzono z użyciem dwóch obiektywnych metryk jakości: WS-PSNR [50], będącej rozszerzeniem miary PSNR dla sekwencji 360°, a także używanej w pracach grupy MPEG VC nad MIV metryki dostosowanej do oceny zniekształceń typowych dla wizji wszechogarniającej – IV-PSNR [51].



■ Rys. 10. Krzywe RD dla sekwencji perspektywicznych (a) i 360° (b); zależność przedstawiono jako IV-PSNR [dB] w funkcji przepływności bitowej [Mbit/s]

Dla obu testowanych profili MIV zastosowano praktyczne ograniczenia liczby i rozmiaru atlasów (4 atlasy, każdy z nich o rozdzielczości do 8 Mpix). W przypadku podejścia *multiview simulcast*, w którym są kodowane widoki wejściowe (o różnej rozdzielczości), zastosowano jedno ograniczenie – łączna liczba przesyłanych pikseli była mniejsza niż  $4 \times 8$  MPix.

### Porównanie profili MIV

Porównanie trzech testowanych podejść ilustruje rys. 10 w formie krzywych **RD** (*Rate-Distortion*), obrazujących zależność jakości sekwencji prezentowanej widzowi od przepływności bitowej zakodowanego strumienia. Przedstawione krzywe zostały uśrednione po wszystkich sekwencjach testowych.

Jak pokazano, w przypadku sekwencji perspektywicznych efektywność kompresji za pomocą koder MIV (obu testowanych profili) jest dużo wyższa, niż dla niezależnego kodowania widoków wejściowych i odpowiadających im map głębi.

Dla systemów o większej przepływności bardziej efektywne jest użycie profilu MIV Main. MIV GA sprawdza się lepiej dla systemów o mocno ograniczonej przepływności. W takim przypadku atlasy głębi stworzone przez koder MIV, działający w trybie MIV Main, są na tyle mocno kompresowane, że dokładność informacji przestrzennej staje się niezadowolająca. Dlatego wydajniej wystymować ją w odbiorniku.

W przypadku sekwencji 360° profil MIV Main umożliwia diametralną poprawę jakości w porównaniu do podejścia *multiview*

■ Tabela 1. Porównanie efektywności kodowania *multiview simulcast* i dwóch profili MIV przedstawione w postaci delty Bjontegaarda; ujemna wartość delty Bjontegaarda oznacza, iż MIV jest efektywniejszy, niż *multiview simulcast*.

Typ	Nazwa sekwencji	MIV Main a <i>multiview simulcast</i>		MIV GA a <i>multiview simulcast</i>	
		WS-PSNR	IV-PSNR	WS-PSNR	IV-PSNR
		Stworzona komputerowo 360°	Chess	lepiej	lepiej
ChessPieces	lepiej		lepiej	gorzej	-8.6%
ClassroomVideo	-24.6%		-16.2%	129.4%	25.8%
Hijack	-54.1%		-59.5%	gorzej	gorzej
Museum	-18.8%		-25.8%	133.6%	42.3%
Stworzona komputerowo Perspektywiczna	Cadillac	-4.3%	-24.8%	-74.6%	-72.6%
	Fan	-32.5%	-44.4%	-91.8%	-83.7%
	Kitchen	-34.6%	-54.9%	-39.0%	-24.9%
	Mirror	-38.5%	-45.9%	-67.9%	-67.6%
	Carpark	-47.6%	-50.5%	-64.0%	-61.9%
Naturalna Perspektywiczna	Fencing	-32.2%	-33.1%	-38.8%	-54.3%
	Frog	-6.1%	-24.4%	-61.5%	-61.7%
	Hall	-77.6%	-68.3%	-88.2%	-63.7%
	Painter	-16.2%	-29.3%	-68.2%	-58.8%
	Street	-18.8%	-40.7%	-51.1%	-56.9%

*simulcast*. Profil MIV GA wydaje się mniej efektywny, aczkolwiek nie jest to problem samego profilu, lecz ograniczeń istniejących metod estymacji map głębi [46], które aktualnie nie są w stanie wyestymować dobrej jakości map dla takiego rodzaju sekwencji.

Wyniki dla poszczególnych sekwencji przedstawiono w tabeli 1. Dla każdej sekwencji zaprezentowano wartość delty Bjontegaarda [52], określającej procentową zmianę przepływności przy zachowaniu takiej samej jakości, wyliczoną na podstawie odległości pomiędzy dwiema krzywymi RD. Deltę Bjontegaarda wyznaczono dla obu metryk używanych w pracach grupy MPEG VC: WS-PSNR [50] i IV-PSNR [51].

Brak wartości w tabeli 1 (np. dla sekwencji Chess) oznacza, iż krzywe są na tyle odległe od siebie, że niemożliwe jest wyznaczenie dla nich delty Bjontegaarda. W takim przypadku lepszą lub gorszą efektywność kompresji koder MIV opisano słownie.

### DALSZE PRACE

Norma MIV została ukończona w tym roku, a więc wydawać by się mogło, iż prace nad efektywnym kodowaniem wizji wszechgarniającej zostały zakończone z sukcesem.

Jednakże pomimo że MIV umożliwia skuteczną kompresję sekwencji wielowidokowych i ma spory potencjał do bycia normą powszechnie używaną, nie rozwiązuje wszystkich potencjalnych problemów i wyzwań towarzyszących przetwarzaniu wizji wszechgarniającej. Stąd też obecnie trwają prace nad drugą edycją normy MIV [53], która ma rozwijać techniki kompresji zastosowane w MIV, dodatkowo zwiększając jej wszechstronność.

Wśród głównych wyzwań stojących aktualnie przed ekspertami grupy MPEG VC zajmującymi się tworzeniem normy MIV można wymienić :

- efektywną kompresję scen nielambertowskich (duża liczba odbić i przezroczystości),
- możliwość użycia systemów z dynamicznie zmieniającymi się parametrami (ruchome kamery, możliwość dynamicznej zmiany ogniskowej poszczególnych kamer),
- możliwość użycia dodatkowych sensorów głębi umożliwiających szybszą lub lepszą estymację głębi dla pozostałych kamer systemu,
- rozszerzenia w kierunku wirtualnej i rozszerzonej rzeczywistości oraz możliwość wstawiania do sceny zarejestrowanej systemem wielokamerowym obiektów reprezentowanych przez chmury punktów.

Praca finansowana ze środków przyznanych przez Ministerstwo Edukacji i Nauki.

### LITERATURA

- [1] Jeong J.B. i in. 2019. „Towards 3DoF+ 360 video streaming system for immersive media”. *IEEE Access* 7: 136399-136408.
- [2] Tanimoto M. i in. 2012. „FTV for 3-D Spatial Communication”. *Proc. of the IEEE* 100 (4): 905-917.
- [3] Stankiewicz O. i in. 2018. „A free-viewpoint television system for horizontal virtual navigation”. *IEEE Tr. on Multimedia* 20 (8): 2182-2195.
- [4] Wien M. i in. 2019. „Standardization status of immersive video coding”. *IEEE J. on Emerging and Selected Topics in Circuits and Systems* 9 (9).

- [5] Oyman O. i in. 2019. „Virtual Reality Industry Forum's View on State of the Immersive Media Industry”. *SMPTE Motion Imaging Journal* 128 (8): 91-96.
- [6] Hayes C. 2021. „Olympic games broadcasting: Immerse yourself in the olympics this summer”. *Engineering & Technology* 16 (6): 2225.
- [7] Wu F. 2021. *Construction of Digital Dynamic Sports System Platform Based on VR Technology*. CIPAE Conference 2021.
- [8] Alhadad S.A. 2018. „Enhancing Smart Sport Management based on Information Technology”. *IOSR-JSPE* 5 (5): 19-26.
- [9] Xu S. i C.Tang. 2021. *Application of AI technology in sports competitions*. AIVRV 2021, Sanya, Chiny.
- [10] Sullivan G. i in. 2012. „Overview of the High Efficiency Video Coding (HEVC) standard”. *IEEE Tr. on Circuits and Systems for Video Technology* 22 (12): 1649-1668.
- [11] ISO/IEC. 2022. Standard ISO/IEC DIS 23090-12, Information technology – Coded Representation of Immersive Media – Part 12: MPEG immersive video.
- [12] Müller K. i in. 2011. „3-D Video Representation Using Depth Maps”. *Proc. of the IEEE* 99 (4): 643-656.
- [13] ISO/IEC. 2021. Standard ISO/IEC 23090-5, Information technology – Coded Representation of Immersive Media – Part 5: Visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC).
- [14] ang E.S. i in. 2019. „Video-Based Point-Cloud-Compression Standard in MPEG: From Evidence Collection to Committee Draft [Standards in a Nutshell]”. *IEEE Signal Processing Magazine* 36 (3): 118-123.
- [15] <https://learnopencv.com/camera-calibration-using-opencv/>
- [16] Szeliski R. 2022. *Computer Vision: Algorithms and Applications*. The University of Washington.
- [17] Dziembowski A., M. Domański. 2018. *Adaptive color correction in virtual view synthesis*. 3DTV-CON, Helsinki, Finlandia.
- [18] Dziembowski A. i in. „Color correction for immersive video applications”. *IEEE Access* 9: 75626-75640.
- [19] Laga H. i in. 2022. „Survey on Deep Learning Techniques for Stereo-Based Depth Estimation.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4): 1738-1764.
- [20] Mieloch D. i in. 2021. „Depth Map Refinement for Immersive Video”. *IEEE Access* 9: 10778-10788.
- [21] Fachada S. i in. 2018. *Depth image based view synthesis with multiple reference views for virtual reality*. 3DTV-CON, Helsinki, Finlandia.
- [22] ISO/IEC. 2019. *Versatile view synthesizer (VVS) 2.0 manual*. Dok. ISO/IEC JTC 1/SC29/WG11 MPEG N18172.
- [23] Dziembowski A. i in. 2019. *Virtual view synthesis for 3DoF+ video*. PCS 2019, Ningbo, Chiny.
- [24] Stankowski J., A. Dziembowski. 2022. *Real-time CPU-based view synthesis for omnidirectional video*. WSCG 2022, Pilzno, Czechy.
- [25] Fleureau J. i in. 2020. *An Immersive Video Experience with Real-Time View Synthesis Leveraging the Upcoming MIV Distribution Standard*. ICMEW 2020.
- [26] Stankowski J., A. Dziembowski. 2020. *Fast view synthesis for immersive video systems*. WSCG 20, Pilzno, Czechy.
- [27] „2021 TV Video Specifications”. <https://developer.samsung.com/smarttv/develop/specifications/media-specifications/2021-tv-video-specifications.html>.
- [28] „H.264/H.265 Video Codec Unit” <https://www.xilinx.com/products/intellectual-property/v-vcu.html#overview>.
- [29] Merkle P. i in. 2006. *Efficient Compression of Multi-View Video Exploiting Inter-View Dependencies Based on H.264/MPEG4-AVC*. ICME 2006, Toronto, Kanada.
- [30] Tech G. i in. 2016. „Overview of the Multiview and 3D Extensions of High Efficiency Video Coding”. *IEEE Tr. on Circuits and Systems for Video Technology* 26 (1): 35-49.
- [31] Boyce J. i in. 2021. „MPEG Immersive Video Coding Standard”. *Proc. of the IEEE* 109 (9): 1521-1536.
- [32] Grzelka A. i in. 2022. *The study of the video encoder efficiency in decoder-side depth estimation applications*. WSCG 2022, Pilzno, Czechy.
- [33] Bross B. i in. 2021. „Overview of the Versatile Video Coding (VVC) standard and its applications”. *IEEE Tr. on Circuits and Systems for Video Technology* 31 (10): 3736-3764.
- [34] Sullivan G., T. Wiegand. 2005. „Video compression – from concepts to the H.264/AVC standard”. *Proc. of the IEEE* 93.
- [35] Szekielda J. i in. 2021. *The influence of coding tools on immersive video coding*. WSCG 2021, Pilzno, Czechy.
- [36] Samelak J. i in. 2021. *Efficient immersive video compression using Screen Content Coding*. WSCG 2021, Pilzno, Czechy.
- [37] ISO/IEC. 2021. *Test Model 11 for MPEG Immersive Video*. Dok. ISO/IEC JTC1/SC29/WG04 MPEG VC N0142, Online.
- [38] Kaufman L., P.J. Rousseeuw. 1990. *Partitioning Around Medoids (Program PAM)*. Wiley Series in Probability and Statistics, Hoboken, USA: 68-125.
- [39] Mieloch D. i in. 2022. *Color-dependent pruning in immersive video coding*. WSCG 2022, Pilzno, Czechy.
- [40] Jylänki J. 2010. *A thousand ways to pack the bin—A practical approach to two-dimensional rectangle bin packing*. Tech. Rep., Online.
- [41] Dziembowski A. i in. 2022. *Spatiotemporal redundancy removal in immersive video coding*. WSCG 2022, Pilzno, Czechy.
- [42] Garus P. i in. 2022. „Immersive Video Coding: Should Geometry Information Be Transmitted as Depth Maps?”. *IEEE Tr. on Circuits and Systems for Video Technology* 32 (5): 3250-3264.
- [43] Mieloch D. i in. 2022. „Overview and Efficiency of Decoder-Side Depth Estimation in MPEG Immersive Video”. *IEEE Tr. on Circuits and Systems for Video Technology, Early Access*.
- [44] Szydełko B. i in. 2022. „Recursive block splitting in feature-driven decoder-side depth estimation”. *ETRI Journal* 44 (1): 38-50.
- [45] ISO/IEC. 2021. *Manual of IVDE 3.0*. Dok. ISO/IEC JTC1/SC29/WG4 MPEG VC/N0058.
- [46] Mieloch D. i in. 2020. „Depth Map Estimation for Free-Viewpoint Television and Virtual Navigation”. *IEEE Access* 8: 5760-5776.
- [47] Mieloch D. i in. 2021. *Point-to-Block Matching in Depth Estimation*. WSCG 2021, Pilzno, Czechy.
- [48] ISO/IEC. 2022. *Common Test Conditions for MPEG Immersive Video*. Dok. ISO/IEC JTC 1/SC 29/WG04 N0169.
- [49] Wieckowski A. i in. 2021. *VVenC: an open optimized VVC encoder in versatile application scenarios*. *Proc. SPIE* 11842, App. of Digital Image Proc. XLIV.
- [50] Sun Y. i in. 2017. „Weighted-to-spherically-uniform quality evaluation for omnidirectional video”. *IEEE Signal Processing Letters* 24 (9): 1408-1412.
- [51] Dziembowski A. i in. 2022. „IV-PSNR – the objective quality metric for immersive video applications”. *IEEE Tr. on Circuits and Systems for Video Technology, Early Access*.
- [52] Bjoentegaard G. 2001. *Calculation of average PSNR differences between RD-Curves*. ITU-T VCEG Meeting, Austin, USA.
- [53] ISO/IEC. 2022. *Use cases and requirements for MIV – edition-2 (final)*. Dok. ISO/IEC JTC 1/SC 29/WG02 N0157.