

Grafika Komputerowa i Multimedia

Wykład 14 Kompresja wideo

Damian Grela
e-mail: dgrela@pk.edu.pl
<http://www.dgrela.pl>

```
static public String show...  
String str = Integer.toString(456);  
int count = leadingZeros(count - str.length());  
int count = leadingZeros(count + 1);
```

```
begin  
process  
variable X : int; begin  
X := 2 and 0;  
after 10;  
end
```

- MPEG-1
- MPEG-2
- Inne
 - MPEG-3
 - MPEG-4
 - MPEG-7

- Charakterystyka cyfrowych danych wideo

- standard CCIR 601 (pierwszy ogólnościatowy standard, profesjonalny sprzęt wideo, cyfrowa TV)

- 720x480 (typowo luminancja, chrominancja 360x480),
60 półobrazów/s (NTSC)

- 720x576, 50 półobrazów/s (PAL)

- piksel zapisywany w przestrzeni barw YUV (po 8 bitów na składową)

- $\approx 1\text{MB}/\text{obraz}$, ponad 30 MB/s

- CD-ROM zmieści około 20s nieskompresowanego wideo w tym formacie

- **Charakterystyka cyfrowych danych wideo**

- standard CIF (wideokonferencje)

- luminancja 352x288, chrominancja 176x144

- przestrzeń barw YUV

- ≈ 150 kB/obraz

- QCIF — $\frac{1}{4}$ rozdzielczości CIF

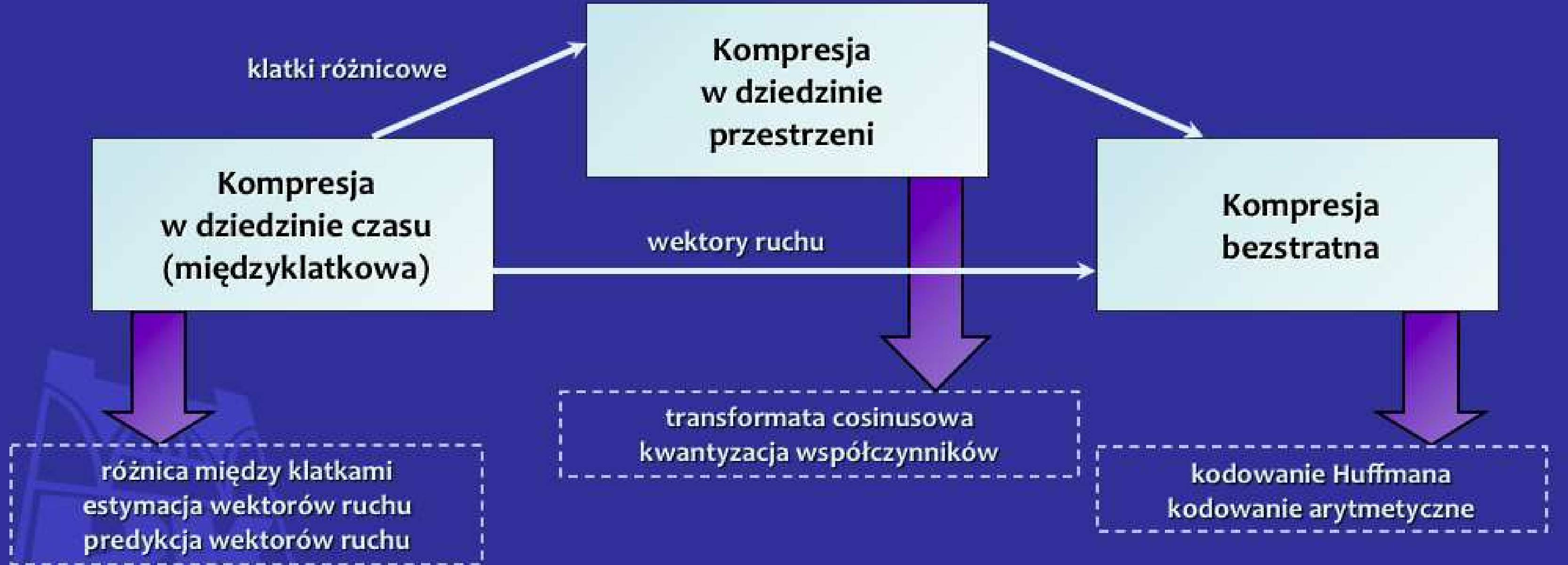
- objętość właściwie wymusza stosowanie kompresji stratnej

- Charakterystyka cyfrowych danych wideo
 - podobnie jak w obrazach statycznych akceptujemy zniekształcenia obrazu i mniejszą rozdzielczość składowych chrominancji
 - ale dla obrazów ruchomych próg tolerancji jest znacznie większy — zatem dopuszczamy znacznie większe straty jakości obrazu po dekompresji
 - pogorszenie jakości zauważymy robiąc „stop-klatkę”
 - kolejne klatki obrazu są silnie skorelowane
 - niekiedy prawie identyczne
 - problem: korelacja często silniejsza względem następnej ramki, niż względem poprzedniej
 - korelacja części obszarów do poprzedniej, a części do następnej ramki

Schemat kompresora video

```
public String show() {  
    String str = Integer.toString(TABLE);  
    int count = leadingZeros(count - str.length());  
    return str + "\n";  
}
```

```
begin  
variable X : int;  
X := 7 and 0;  
after 100  
end
```



- Opracowany przez Moving Picture Experts Group w 1992 (prace rozpoczęto w 1988), ISO IS 11172
 - główne przeznaczenie: składowanie wideo (na CD-ROM)
 - obraz wideo bez przeplotu 352x240, 30 obrazów/s (NTSC) lub 352x288, 25 obrazów/s (PAL, SECAM)
 - kompresja obrazów z przepływnością do 1.5 Mb/s
 - kompresja dźwięku z przepływnością do 192 kb/s
 - algorytm silnie niesymetryczny
 - pierwotnie kodowanie off-line lub wspomagane sprzętowo

- **Reprezentacja danych obrazowych w MPEG-1**

- ciąg makrobloków, z których każdy zawiera:

- 4 bloki 8x8 pikseli składowej luminancji

- po jednym bloku 8x8 dla składowych chrominancji

- makrobloki grupowane są w warstwy

- warstwa zawiera kolejne (w porządku rastrowym) makrobloki o takim samym poziomie jasności

- warstwy tworzą obraz, obrazy grupy a następnie ciąg obrazów, który łączony jest w pakiety z danymi audio

- **Rodzaje obrazów w MPEG-1**

- **obrazy I (intra-coded)**

- kodowane niezależnie od pozostałych, za pomocą DCT (ramka poddana wyłącznie kompresji przestrzennej)
 - umożliwiają swobodny dostęp do danych wideo

- **obrazy P (predictive-coded)**

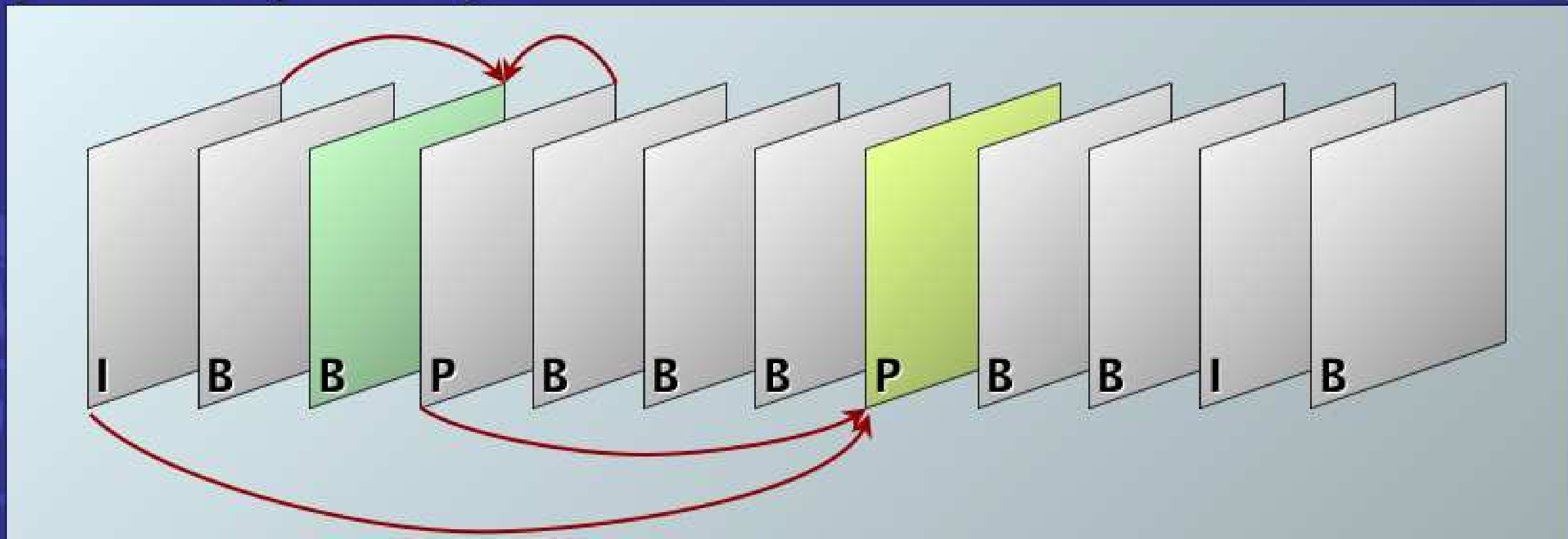
- kodowane również z pomocą DCT, ale po przeprowadzeniu predykcji na podstawie poprzedniego obrazu I lub P

- **obrazy B (bidirectionally predictive-coded)**

- predykcja na podstawie poprzedniego i następnego obrazu I lub P
 - dopuszczalna niska jakość kompresji (błędy nie propagują się)

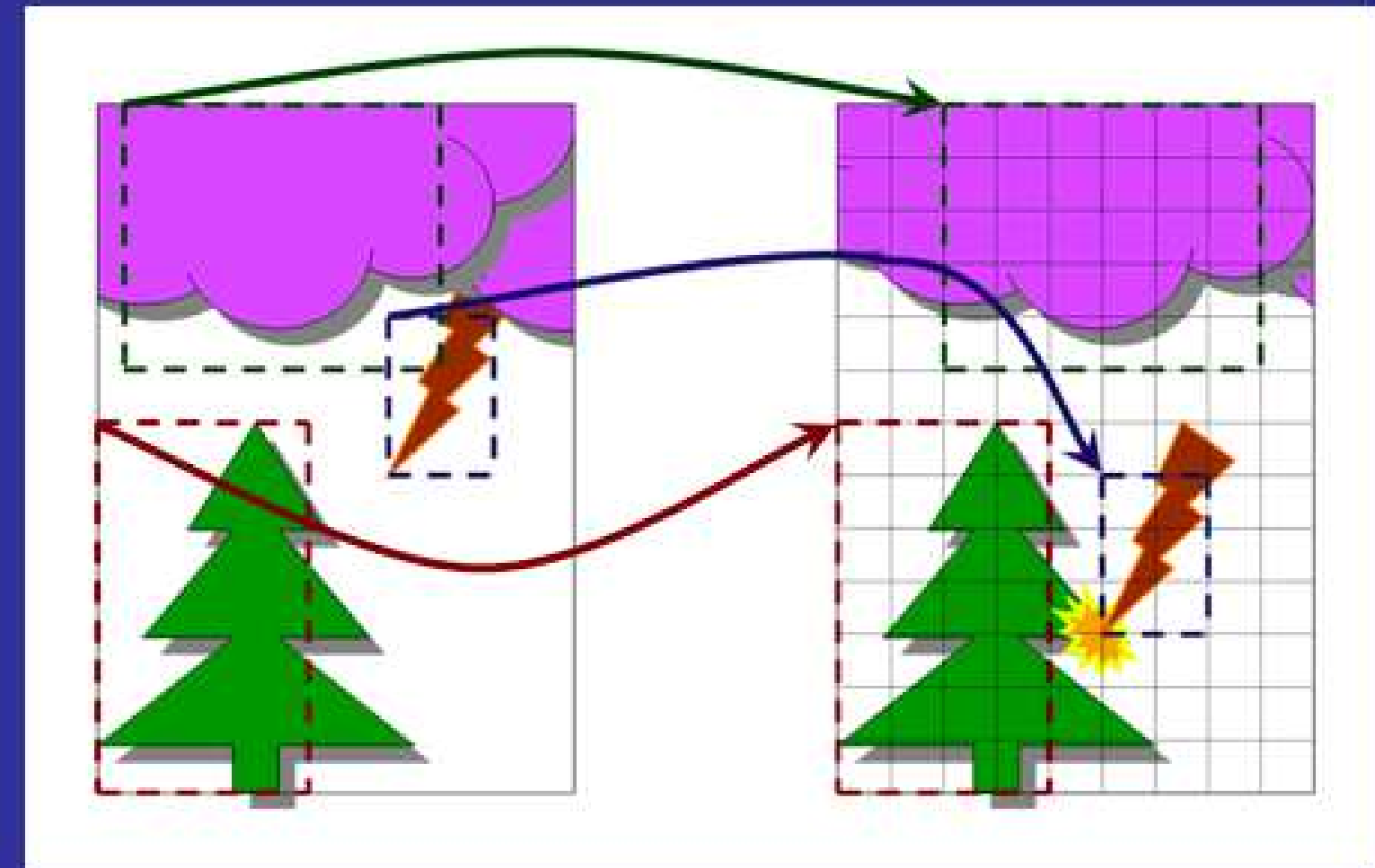
- Sekwencja obrazów I, P i B (typowa)

- UWAGA: transmisja w odmiennej kolejności; a dekodery musi mieć bufor obrazów I i P jeszcze nie wyświetlonych



- **Pretykcja:**
Estymacja ruchu

- w MPEG-1 wyszukujemy jedynie translacje
(można by obrót, skalowanie itp.
— jak w kompresji fraktalnej)



- **Predykcja:**
Estymacja ruchu
 - dla danego makrobloku znajdujemy w obrazie odniesienia blok najbardziej do niego zbliżony
 - standard definiuje jedynie sposób dekodowania strumienia MPEG
 - zazwyczaj estymację ruchu przeprowadzamy na podstawie składowej luminancji

- **Predykcja:**
 - Estymacja ruchu**
 - przeprowadzenie dopasowywania należy do kodera
 - używamy algorytmów estymacji ruchu
 - porównanie wszystkich/prawie wszystkich makrobloków obrazu odniesienia (maksymalnie 396 dla MPEG-1) z danym makroblokiem kodowanego obrazu (np. na podstawie błędu średniokwadratowego) byłoby zbyt złożone

- **Predykcja na bazie estymacji ruchu**

- dla obrazów P

- zapamiętujemy współrzędne dopasowanego bloku, lub informację, że nie udało się znaleźć zadowalającego dopasowania

- (w takim przypadku predykcja nie jest przeprowadzana)



–dla obrazów B

- znajdujemy i zapamiętujemy współrzędne dopasowanego bloku w najbliższym poprzednim oraz najbliższym następnym obrazie I lub P, predykcja na podstawie:
 - dopasowania w poprzednim obrazie I lub P,
 - w następnym obrazie I lub P,
 - interpolacji (uśrednienia) dopasowań z obu obrazów
 - bez predykcji gdy oba dopasowania niezadawalające



• Kodowanie obrazów w MPEG-1

– Kodowanie odbywa się w sposób bardzo zbliżony do JPEG

- dla obrazów P i B po przeprowadzeniu predykcji kodujemy obraz błędu predykcji składowych, dla obrazów I wprost składowe pikseli
- używamy odmiennej tablicy kwantyzacji dla obrazów I, innej dla P oraz B
- zmieniając mnożnik dla tablic kwantyzacji sterujemy wartością bitrate
- dla każdego makrobloku kodujemy które bloki 8x8 pikseli zawierają współczynniki niezerowe
- używamy predefiniowanych kodów Huffmana



- Rozbicie sygnału na składową luminancyjną i dwie chrominancyjne ($Y C_r C_b$).
- Zmniejszenie rozdzielczości chrominancji (4:2:0, a więc x2 w pionie i w poziomie).
- Zastosowanie transformaty DCT do koncentracji energii sygnału w małej ilości współczynników.
- Zbieranie współczynników DTC zygzakiem, co koncentruje zera na końcu.
- Kwantyzacja współczynników, zastosowanie różnych prógów dla różnych współczynników.
- Kompresja Huffmana na końcu.

- Należy jednak pamiętać o istotnych różnicach:
 - Transformacie poddawane są nie bloki obrazu, a bloki obrazu „resztkowego” po usunięciu danych zawartych w wektorach predykcji ruchu.
 - To może powodować inny rozkład widmowy energii -> inny rozkład częstości współczynników DCT -> inne alfabetory kodowania Huffmana.

- **Opracowany przez Moving Picture Experts Group w 1995**

- **standard ogólniejszy od MPEG-1**

- **składowanie wideo (CD/DVD inne nośniki)**
- **transmisja wideo (TV, TV-sat, internet)**



– standard ogólniejszy od MPEG-1

- różne rozdzielczości obrazu
- różne częstotliwości wyświetlania
- kodowanie sekwencji wideo jako obrazów z przeplotem lub bez
 - dla przeplotu dodatkowe tryby predykcji oraz estymacji ruchu

– podział koderów MPEG-2 na klasy

- ze względu na poziomy i profile zdefiniowane w standardzie (następne slajdy)
- klasa definiuje możliwości kodera i wymagania dla dekodera

(za: Sayood: Kompresja danych. Wprowadzenie, RM, W-wa, 2002)

• Profile w MPEG-2

- definiują ograniczenia/wymagania odnośnie algorytmów (de)kompresji

- profil wyższy jest kompatybilny w dół z profilem niższym

prosty → główny → snr-skalowalny → przestrzennie-skalowalny → wysoki

- profil prosty — najmniejsze wymagania (m.in. bez obrazów B)
- profil główny odpowiada MPEG-1
- profile snr-skalowalny, przestrzennie-skalowalny i wysoki generują kilka strumieni skompresowanego wideo,
 - strumień bazowy ma niską jakość i przepływność,
 - dodatkowe strumienie polepszają jakość (np. snr)
 - mając wideo skompresowane w MPEG-2 możemy bez dekodowania transmitować dane do odbiorców dysponujących łączem o niskiej przepustowości (tylko strumień bazowy), dla lepszego łącza transmitujemy również strumienie dodatkowe

- **Poziomy w MPEG-2**

- definiują ograniczenia/wymagania odnośnie parametrów strumienia wideo

- poniżej przykładowe górne ograniczenia poziomów w MPEG-2

- (za: Drozdek: Wprowadzenie do kompresji danych, WNT, W-wa 1999)

Poziom	Próbki/ wiersz	Wiersze/ obraz	Obrazy/s	Próbki luminancji/s	Mb/s
Wysoki	1920	1152	60	62668800	80
Wysoki typu 1440	1440	1152	60	47001600	60
Główny	720	576	30	10368000	15
Niski	352	288	30	3041280	4

• Poziomy i profile w MPEG-2

– dopuszczalne tylko niektóre kombinacje

Poziom	Profil prosty	Profil główny	Profil SNR-skalowalny	Profil przestrzennie-skalowalny	Profil wysoki
Wysoki	nie	tak	nie	nie	tak
Wysoki typu 1440	nie	tak	nie	tak	tak
Główny	tak	tak	tak	nie	tak
Niski	nie	tak	tak	nie	nie

- Przyjęło się uważać, że video MPEG-1 ma rozmiary do 352 x 288 pikseli, nie jest to do końca prawda, standard dopuszcza do 4096 x 4096 pikseli, oczywiście przy znacznie większym bitrate niż standardowe 1536 kbit/s.

- MPEG-2 pozwala dodatkowo na:
 - Kompresję obrazów z przeplotem.
 - Inne sposoby zbierania współczynników DCT oprócz zygzaka.
 - Większy bitrate (większa rozdzielczość, lepsza jakość obrazu).
 - Dźwięk wielokanałowy (MPEG-1 ma tylko stereo).
 - Możliwość dekodowania do niepełnej rozdzielczości.
 - Możliwość dekodowania z niepełną szybkością (np. co drugą ramkę).

• MPEG-4

- nacisk na zwiększoną funkcjonalność, zastosowania jak: multimedialne bazy danych, usługi interakcyjne, urządzenia „embedded”, integracja z systemami miksującymi/reenderującymi obraz, generującymi animacje i in.
- rozbitcie danych wideo, tzw. sceny na obiekty takie, jak np. nieruchome tło, dźwięk, ruchome postacie lub napisy poruszająca się na kodowanym oddzielnie tle

– wiele typów obiektów

- obiekty wideo ale nie prostokątne (kanał alfa, maskowanie)
- nowe algorytmy kodowania mowy (m.in. CELP) i dźwięku (syntezowanie dźwięku, MIDI, wavetable)
- reprezentacja obiektów z pomocą modeli zbudowanych z siatek i tekstur (zastosowanie do animacji twarzy i postaci); tekstura może być również strumieniem wideo
- scenę opisuje język BIFS, bazujący na VRML
- kodowanie obiektów w oddzielnych strumieniach
- „skalowanie obiektowe” — zmniejszenie jakości przez transmisję tylko istotnych obiektów
- pozycjonowanie obiektów w przestrzeni 3D (i obraz i dźwięk)

- **MPEG-3**

- przeznaczony dla kodowania transmisji HDTV
- opracowywanie standardu przerwano, gdy okazało się, że MPEG-2 spełnia większość założeń MPEG-3

- **MPEG-7**

- to nie jest algorytm kompresji wideo, standard definiuje sposób opisu informacji multimedialnej dla potrzeb wyszukiwania, indeksowania, filtrowania itp.



Dziękuję za uwagę...

```
static public String show...
String str = Integer.toString(155);
int count = leadingZeros(count - str.length());
int count = leadingZeros(count + 1);
```

```
process
variable X : int;
begin
X := 2 and 0;
... after 70;
```



Damian Grela
e-mail: dgrela@pk.edu.pl
<http://www.dgrela.pl>

