

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

MODERNÍ VIDEO KODEKY

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

JAN BÍLEK

BRNO 2015



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

## **MODERNÍ VIDEO KODEKY**

MODERN VIDEO CODECS

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**JAN BÍLEK**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. PETR ČÍKA, Ph.D.**

BRNO 2015



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Bakalářská práce

bakalářský studijní obor  
Teleinformatika

**Student:** Jan Bílek

**ID:** 152377

**Ročník:** 3

**Akademický rok:** 2014/2015

**NÁZEV TÉMATU:**

## Moderní video kodeky

### POKYNY PRO VYPRACOVÁNÍ:

Prostudujte moderní kompresní formáty digitálního videa, zejména H.264 AVC, H.265 HEVC, VP8, VP9, způsoby objektivního i subjektivního vyhodnocení kvality videa a druhy chyb, které se mohou vyskytnout během streamování videa přes Internet. Navrhněte aplikaci, která umožní kompresi a dekompresi vstupního videa dle výše zmíněných standardů; simulaci přenosového kanálu včetně zanesení chyb; vyhodnocení kvality výsledného videa. Aplikace bude mít grafické uživatelské rozhraní s intuitivním ovládáním. Práce bude dále obsahovat porovnání kompresních standardů.

### DOPORUČENÁ LITERATURA:

[1] RICHARDSON, Iain E. The H.264 Advanced Video Compression Standard. London: Wiley-Blackwell, 2010. ISBN 978-0470516928.

[2] WebM: an open web media project [online]. 2010-2013 [cit. 2013-10-16]. Dostupné z: <http://www.webmproject.org/>

**Termín zadání:** 9.2.2015

**Termín odevzdání:** 2.6.2015

**Vedoucí práce:** Ing. Petr Číka, Ph.D.

**Konzultanti bakalářské práce:**

**doc. Ing. Jiří Mišurec, CSc.**

*Předseda oborové rady*

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Tato bakalářská práce se zabývá moderními video kodeky a hodnocením jejich výstupní kvality videa. Nejprve jsou vysvětleny principy některých objektivních i subjektivních metod hodnocení. V další části jsou poté představeny nejpoužívanější video kodeky, konkrétně H.264, H.265, VP8, VP9, Dirac 2.2.3, WMV a Sorenson Spark. Výstupem práce je desktopová aplikace VideoCodecs napsaná v jazyce C++, která pro kódování a dekódování používá knihovnu ffmpeg. Součástí aplikace jsou také algoritmy PSNR, SSIM a M-SVD pro objektivní hodnocení videa, dále pak simulace přenosového kanálu, umožňující zanesení chyb. Pomocí této aplikace bylo provedeno měření na čtyřech testovacích sekvencích v HD rozlišení při různých bitových rychlostech. Ze srovnání jednotlivých kodeků vyplývá, že nejlepším kodekem, s ohledem na kvalitu obrazu, je kodek H.265 před kodekem H.264, VP9 a VP8.

## **KLÍČOVÁ SLOVA**

Komprese videa, M-SVD, Objektivní kvalita, PSNR, Srovnání kodeků, SSIM, Video kodeky

## **ABSTRACT**

This bachelor's thesis focuses on modern video codecs and its output video quality assessment. Firstly, principles of some objective and subjective assessment methods are explained. Afterwards, the most common video codecs are introduced, namely H.264, H.265, VP8, VP9, Dirac 2.2.3, WMV and Sorenson Spark. The outcome of this thesis is a desktop application named VideoCodecs written in C++ language, which uses ffmpeg library for encoding and decoding tasks. As a part of the application, algorithms PSNR, SSIM and M-SVD for objective video quality assessment were implemented. Furthermore, a simulation of transport channel was included, enabling us to bring errors into provided video. Using this application, video quality measurements were carried out, on four test sequences in HD resolution at several bitrates. According to our results, the best video codec regarding the video quality is H.265 before H.264, VP9 and VP8.

## **KEYWORDS**

Codec Comparison, M-SVD, Objective Quality, PSNR, SSIM, Video Codecs, Video Compression

BÍLEK, Jan *Moderní video kodeky*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 53 s. Vedoucí práce byl Ing. Petr Číka, PhD



## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Moderní video kodeky“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora(-ky)

## PODĚKOVÁNÍ

Rád bych poděkoval panu Ing. Petru Číkovi, Ph.D. za odborné vedení práce a cenné metodické rady při jejím zpracování.

Brno .....

.....

podpis autora(-ky)



Faculty of Electrical Engineering  
and Communication  
Brno University of Technology  
Purkynova 118, CZ-61200 Brno  
Czech Republic  
<http://www.six.feec.vutbr.cz>

## PODĚKOVÁNÍ

Výzkum popsany v této bakalářské práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno .....

.....

podpis autora(-ky)



EVROPSKÁ UNIE  
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ  
INVESTICE DO VAŠÍ BUDOUCNOSTI



# OBSAH

Úvod	11
<b>1 Metody hodnocení kvality videa</b>	<b>12</b>
1.1 Objektivní metody	12
1.1.1 PSNR	12
1.1.2 SSIM	13
1.1.3 M-SVD	15
1.1.4 DVQ	16
1.1.5 PSNR-H	17
1.1.6 PVQM	17
1.1.7 VIF	18
1.1.8 VSNR	18
1.2 Subjektivní metody	19
1.2.1 ACR	19
1.2.2 ACR-HR	20
1.2.3 DCR	20
1.2.4 PC	21
1.2.5 CCR	21
1.2.6 SSCQE	22
1.2.7 DSCQS	22
<b>2 Zkoumané video kodeky</b>	<b>23</b>
2.1 H.264	23
2.2 H.265	26
2.3 VP8	27
2.4 VP9	28
2.5 Dirac	28
2.6 WMV	29
2.7 Sorenson Spark	30
<b>3 Druhy chyb</b>	<b>31</b>
<b>4 Praktická část</b>	<b>32</b>
4.1 Realizace	32
4.2 Metody	35
4.3 Výsledky	38
<b>5 Závěr</b>	<b>45</b>

Literatura	46
Seznam symbolů, veličin a zkratk	50
Seznam příloh	52
A Obsah Přiloženého CD	53

## SEZNAM OBRÁZKŮ

1.1	Postup při výpočtu DVQ. [31]	16
1.2	Škála pro hodnocení metodou SSCQE [11]	22
2.1	Obecný postup kódování H.264. [17]	24
2.2	Obecný postup dekódování H.264. [17]	25
4.1	Hlavní okno programu VideoCodecs.	33
4.2	Okno <i>Encode</i> pro kódování videa do požadovaných formátů.	34
4.3	Okno <i>Test</i> pro měření kvality videa.	35
4.4	Okno <i>Channel</i> sloužící pro simulaci přenosového kanálu.	35
4.5	Obecný postup hodnocení kvality videa plně referenční metodou.	36
4.6	Ukázkový snímek z videa Mobcal	36
4.7	Ukázkový snímek z videa Blue sky	37
4.8	Ukázkový snímek z videa Life	37
4.9	Ukázkový snímek z videa Old town	37
4.10	Mobcal – průměrné hodnoty PSNR v závislosti na bitové rychlosti.	39
4.11	Mobcal – hodnoty MSSIM v závislosti na bitové rychlosti.	39
4.12	Mobcal – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.	40
4.13	Blue sky – průměrné hodnoty PSNR v závislosti na bitové rychlosti.	40
4.14	Blue sky – hodnoty MSSIM v závislosti na bitové rychlosti.	41
4.15	Blue sky – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.	41
4.16	Life – průměrné hodnoty PSNR v závislosti na bitové rychlosti.	42
4.17	Life – hodnoty MSSIM v závislosti na bitové rychlosti.	42
4.18	Life – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.	43
4.19	Old town – průměrné hodnoty PSNR v závislosti na bitové rychlosti.	43
4.20	Old town – hodnoty MSSIM v závislosti na bitové rychlosti.	44
4.21	Old town – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.	44

## SEZNAM TABULEK

1.1	Tabulka kategorií ACR . . . . .	19
1.2	Tabulka kategorií DCR . . . . .	21
1.3	Tabulka kategorií CCR . . . . .	21

# ÚVOD

Moderní technologie umožňují zobrazovat videa ve stále vyšším rozlišení na všech možných zařízeních. Aby bylo možné tato objemná data spolehlivě přenášet uživatelům pomocí komunikačních technologií, které v současné době nedosahují potřebných kapacit, je nutné vyvíjet stále lepší video kodeky. Takové, které podle lidského zraku zachovají maximální kvalitu obrazu při co nejnižší bitové rychlosti. Za tímto účelem vyvinula v posledních letech organizace JCT-VC (Joint Collaborative Team on Video Coding) kodek H.265/HEVC, jako nástupce kodeku H.264/AVC, a firma Google kodek VP9, který má překonat svého předchůdce VP8, a současně vytvořit vhodnou alternativu pro kodek H.265.

Abychom byli schopni kódovaná videa objektivním a efektivním způsobem porovnat, je třeba nejprve definovat algoritmy pro hodnocení obrazové kvality videa. Hlavním zájmem těchto algoritmů je, aby co možná nejvíce korelovaly se subjektivním hodnocením kvality obrazu, v nejlepším případě dokázaly automaticky vypočítat subjektivní skóre, a zároveň nebyly neúměrně složité pro výpočet. V této bakalářské práci si nejprve představíme některé takové algoritmy a v další části se zaměříme i na metody subjektivního hodnocení. Následně se podrobněji seznámíme s výše uvedenými a některými dalšími moderními kodeky.

Výsledkem této práce je desktopová aplikace s grafickým uživatelským rozhraním, díky níž je možné kódovat a dekódovat video užitím všech uvedených kodeků. Pro praktická měření obrazové kvality videa jednotlivých kodeků byly zároveň implementovány tři vybrané algoritmy PSNR, SSIM a M-SVD, pomocí kterých budou všechny popsané kodeky porovnány.



# 1 METODY HODNOCENÍ KVALITY VIDEOA

Metody hodnotící kvalitu komprimovaného videa lze podle [29] rozdělit několika způsoby. Z hlediska přítomnosti lidského faktoru rozlišujeme metody subjektivní a metody objektivní. Subjektivní metody jsou obecně velice spolehlivé, jelikož představují skutečně vnímanou kvalitu videa. Jejich nevýhodou je časová náročnost, plynoucí z nutnosti zapojení velkého počtu pozorovatelů pro dosažení reprezentativních výsledků. Objektivní metody měří kvalitu videa užitím daných algoritmů. Jejich cílem je se co nejvíce přiblížit výsledkům subjektivního hodnocení, což by umožnilo provádět spolehlivá měření obrazové kvality videa v relativně krátkém čase.

## 1.1 Objektivní metody

Hlavním hlediskem pro rozdělení objektivních hodnotících metod je užití původního nekomprimovaného videa. Metody, které k měření využívají jak původní, tak komprimované video, se nazývají referenční. Ty lze dále rozdělit na plně referenční a částečně referenční. Nereferenční metody ke svému hodnocení původní video vůbec nevyužívají, což může být v určitých případech žádoucí. Následně budou popsány metody, které mohou být použity k objektivnímu hodnocení videa.

### 1.1.1 PSNR

Špičkový poměr signálu k šumu – Peak Signal-to-Noise Ratio, je velice často užívaná plně referenční metoda. Její velká výhoda tkví v jednoduchosti implementace, a tedy i nízké náročnosti jejího výpočtu [27]. Při výpočtu PSNR je nejprve zjištěna střední kvadratická chyba – Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2, \quad (1.1)$$

kde  $N$  je celkový počet pixelů a  $x_i$  a  $y_i$  jsou jednotlivé pixely původního a konečného signálu. Výsledný poměr signálu k šumu poté dostaneme jako

$$\text{PSNR} = 10 \log_{10} \frac{(L - 1)^2}{\text{MSE}}, \quad (1.2)$$

kde  $L$  je dynamický rozsah hodnoty pixelu. U videa je to nejčastěji 8 bitů, tedy 256. Pokud se oba signály shodují, MSE je rovno nule, a tedy PSNR není definováno. Za identické jsou proto signály považovány tehdy, je-li PSNR rovno 100 dB. V případech, kdy je PSNR vyšší, je za správný výsledek považována právě tato hodnota.

Použití PSNR jako metriky pro hodnocení kvality obrazu v určitých případech neodpovídá lidskému vnímání, což lze považovat za zásadní nedostatek. [27] Příkladem mohou být strukturální změny, jako například kvantizační či jiný šum, oproti změnám jasu, kontrastu nebo například posunu obrazu o několik pixelů určitým směrem. Druhou skupinu chyb lidské oko nevnímá příliš rušivě, avšak PSNR při nich vykazuje značnou neshodu.

### 1.1.2 SSIM

Metoda SSIM neboli Structural Similarity Index byla vyvinuta za účelem odstranit nedostatky předchozí metody. Prioritou bylo přiblížení se k psychovizuálnímu modelu lidského vidění (HVS). [26] Vychází z předpokladu, že lidský zrak je citlivý spíše na celkovou strukturu obrazu, nežli na jeho jednotlivé pixely. Z tohoto důvodu byl výpočet indexu rozdělen na tři hlavní části, a to srovnání jasu  $l(\mathbf{x}, \mathbf{y})$ , kontrastu  $c(\mathbf{x}, \mathbf{y})$  a srovnání struktury  $s(\mathbf{x}, \mathbf{y})$ . Podle obecného postupu nejprve vypočteme průměrnou hodnotu jasu pro daný signál

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i. \quad (1.3)$$

Výslednou hodnotu poté odebereme z původního signálu jako  $\mathbf{x} - \mu_x$  a provedeme odhad kontrastu signálu užitím směrodatné odchylky

$$\sigma_x = \left( \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2 \right)^{\frac{1}{2}}. \quad (1.4)$$

Následně jsou oba signály před porovnáním struktury normalizovány dělením směrodatnou odchylkou  $(\mathbf{x} - \mu_x) / \sigma_x$ . Posledním krokem je sloučení všech tří částí do jedné obecné funkce

$$S(\mathbf{x}, \mathbf{y}) = f(l(\mathbf{x}, \mathbf{y}), c(\mathbf{x}, \mathbf{y}), s(\mathbf{x}, \mathbf{y})). \quad (1.5)$$

Konkrétní srovnání jasu provedeme jako

$$l(\mathbf{x}, \mathbf{y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \quad (1.6)$$

kde  $C_1$  je konstanta přidaná pro zachování stability rovnice a vypočteme ji jako

$$C_1 = (K_1 L^2), \quad (1.7)$$

kde  $K_1$  je konstanta a volíme ji jako  $K_1 \ll 1$ .

Srovnání kontrastu se velice podobá předchozímu srovnání luminance

$$c(\mathbf{x}, \mathbf{y}) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \quad (1.8)$$

kde  $C_2$  je konstanta a vypočteme ji jako

$$C_2 = (K_2L^2), \quad (1.9)$$

kde  $K_2$  je konstanta a volíme ji opět jako  $K_2 \ll 1$ , avšak odlišně od  $K_1$ .

Pro další postup je nutné nejprve vypočítat korelaci  $\sigma_{xy}$  obou signálů

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y), \quad (1.10)$$

a až posléze jsme schopni porovnat jejich strukturu

$$s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}, \quad (1.11)$$

kde  $C_3$  je konstanta, kterou dostaneme jako  $C_2/2$ .

Po sloučení všech tří funkcí dostáváme po úpravách výsledný vztah pro výpočet SSIM indexu:

$$\text{SSIM} = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (1.12)$$

Při konkrétní implementaci SSIM indexu je vhodné jednotlivé statistické výpočty provádět v rámci lokálního posuvného okna 8 x 8 pixelů, které postupně projde celý snímek tak, aby se jednotlivé kroky překrývaly. [26] Nevýhodou tohoto postupu jsou vznikající blokové artefakty ve výsledné SSIM mapě. Pokud požadujeme přesnější měření, lze za cenu vyšší složitosti výpočtu použít Gaussovo okno  $w_i$  o doporučené velikosti 11 x 11 pixelů.

Pro celkové zhodnocení kvality jednoho snímku je použita průměrná hodnota SSIM pro všechny pozice okna, neboli MSSIM

$$\text{MSSIM}(\mathbf{X}, \mathbf{Y}) = \frac{1}{M} \sum_{j=1}^M \text{SSIM}(\mathbf{x}_j, \mathbf{y}_j), \quad (1.13)$$

kde matice  $\mathbf{X}$ ,  $\mathbf{Y}$  představují původní a zkreslený snímek,  $M$  je celkový počet pozic zvoleného okna a  $j$  je pořadové číslo pozice.

Na bázi SSIM byla vyvinuta řada dalších metod, které tento postup upravují a vylepšují pro konkrétní účely. Například metoda VSSIM (Video SSIM) [28], která používá váhové koeficienty pro jednotlivá okna i snímky a při výpočtu bere v potaz pohybové vektory. Metoda 3D-SSIM [33] rozšiřující výpočet do časové oblasti tím, že vytváří trojrozměrné bloky o dané velikosti a z nich pomocí kombinačních

funkcí celkový index. Na porovnávání snímků při různých rozlišeních je založena metoda MS-SSIM (Multi-Scale SSIM) [30], při které je výsledná hodnota SSIM rovna produktu všech jednotlivých srovnání luminance, kontrastu a struktury.

### 1.1.3 M-SVD

Základní algebraickou operací metody M-SVD je singulární rozklad matice (Singular Value Decomposition), který vychází z předpokladu, že každou matici lze rozložit na součin tří matic.  $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ . Matice  $\mathbf{U}$  a  $\mathbf{V}$  jsou ortogonální a právě diagonála matice  $\mathbf{S}$  obsahuje singulární hodnoty matice  $\mathbf{A}$ . [25]

Originální a zpracovaný snímek je nejprve převzorkován do podoby 4:4:4 YCbCr. Poté je rozdělen na bloky o velikosti 8 x 8 pixelů. Pro každý blok je vypočten singulární rozklad pro jasovou složku Y. Následně je pro každý tento blok vypočtena celková vzdálenost mezi singulárními hodnotami původního a zkresleného signálu.

$$D_k = \sqrt{\sum_{i=1}^n (s_i - \hat{s}_i)^2}, \quad (1.14)$$

kde  $n$  je rozměr bloku,  $s_i$  je  $i$ -tá singulární hodnota daného bloku nezkresleného signálu a  $\hat{s}_i$  signálu zkresleného. Hodnoty vzdáleností mohou být převedeny do rozsahu 0 až 255 a zaneseny do obrázku v odstínech šedi, čímž dostaneme grafickou podobu této metriky, tzv. mapy zkreslení. Výslednou globální hodnotu M-SVD dostaneme jako:

$$\text{M-SVD} = \frac{\sum_{i=1}^m |D_i - D_{mid}|}{m}, \quad (1.15)$$

kde  $m$  je celkový počet bloků ve snímku,  $D_i$  je výsledná vzdálenost  $i$ -tého bloku a  $D_{mid}$  je medián ze všech vzdáleností  $D_i$  pro daný snímek.

Pro získání globální numerické hodnoty, která více koreluje se subjektivním měřením, lze nejprve pro luminanci použít funkci detekce hran, která vrací hodnotu 1, jestliže je pixel součástí hrany, 0 pokud není. Každému bloku je poté přiřazen index  $B_k$ , vyjadřující přítomnost hran v daném bloku.

$$B_k = \frac{\frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^r b_{ij}}{\frac{1}{rc} \sum_{i=1}^c \sum_{j=1}^n b_{ij}}, \quad (1.16)$$

kde  $b_{ij}$  je binární hodnota po detekci hran v  $i$ -tém řádku a  $j$ -tém sloupci daného bloku. Celkovou vzdálenost  $k$ -tého bloku  $D_k$  poté vypočteme jako:

$$D_k = B_k \sqrt{\sum_{i=1}^n (s_i - \hat{s}_i)^2}, \quad (1.17)$$

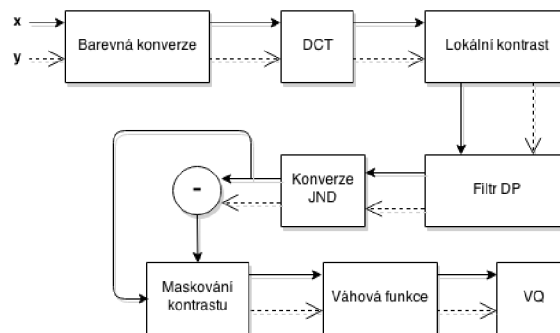
další postup je již shodný s dříve uvedenými kroky. Výpočet M–SVD lze také rozšířit o barevné složky  $C_b$  a  $C_r$ . Pro výpočet globální hodnoty M–SVD použijeme experimentálně zjištěné váhy jednotlivých složek, a to 80% pro M–SVD jasové složky  $Y$  a 10% pro jednotlivé složky chrominanční. Sečtením výsledných hodnot Dostaneme metriku M–SVD/ $YCbCr$ , která je sice přesnější, ale v porovnání s výpočtem pouze pro jasovou složku, koreluje se subjektivními hodnoceními přibližně ve stejné míře, tudíž se ve většině případů nevyplatí počítat i s těmito složkami.

Metoda M–SVD, ve své nejjednodušší verzi bez detekce hran, je přiměřeně náročná na výpočet, na rozdíl od následně uvedených metod a podle autorů dobře koreluje se subjektivním měřením, konkrétně asi o 10% lépe než PSNR.

### 1.1.4 DVQ

Metoda DVQ (Digital Video Quality) je založena na konkrétních měřeních lidského zraku a jeho schopnosti rozpoznat kvantizační šum. Výsledkem těchto měření jsou modelové křivky ve frekvenční oblasti, které udávají práh viditelnosti kvantizačních chyb vzniklých při kvantování koeficientů diskrétní kosinové transformace (DCT). [31]

Obecný postup je zaznamenán na obr. 1.1. Vstupem metriky je referenční videosekvence  $x$  a testovaná sekvence  $y$ . Nejprve je provedena barevná konverze do prostoru YOZ, což je barevný prostor vyvinutý pro usnadnění těchto měření.[15] Poté je provedena DCT na blocích velikosti 8 x 8 pixelů, dalším krokem je výpočet lokálního kontrastu každého bloku. Pomocí filtru typu dolní propust druhého řádu je snížen objem dat nutných k udržení v paměti. Následně je vypočten rozdíl mezi modelovou křivkou a hodnotami DCT koeficientů (Just Noticeable Distortion). Maskování kontrastu referenční sekvence je provedeno jako její dělení rozdílem obou sekvencí. Výsledné koeficienty jsou sečteny pomocí váhové funkce, čímž dostáváme výslednou hodnotu (VQ).



Obr. 1.1: Postup při výpočtu DVQ. [31]

### 1.1.5 PSNR–H

Hlavním cíl metody PSNR–H je, jako u většiny referenčních metod, přiblížení se psychovizuálnímu modelu lidského vidění HVS. Základní vzorec pro výpočet viz 1.2 je stejný jako u metody PSNR, avšak výpočet MSE je proveden naprosto odlišným způsobem, což z PSNR–H činí samostatnou metodu pro hodnocení kvality videa.

Před výpočtem je nejprve sjednocen kontrast a jas obou snímků. Následně je každý snímek procházen oknem o velikosti 64 x 64 pixelů. V každém kroku okna jsou poté transformovány bloky 8 x 8 pixelů pomocí DCT.[5] Výsledné koeficienty jsou kvantizovány užitím kvantizační tabulky formátu JPEG. Hodnotu  $MSE_H$  vypočteme jakou sumu rozdílů jednotlivých transformačních koeficientů umocněných na druhou.

$$MSE_H = K \sum_{i=1}^{I-7} \sum_{j=1}^{I-7} \sum_{m=1}^8 \sum_{n=1}^8 ((X[m, n]_{ij} - X[m, n]_{ij}^e) T_c[m, n])^2, \quad (1.18)$$

kde koeficienty I, J představují rozměr obrázku,  $K = 1/[(I - 7)(J - 7)64]$ ,  $X_{ij}$  a  $X_{ij}^e$  jsou zkreslená a původní matice, obsahující DCT koeficienty jednotlivých bloků o velikosti 8 x 8 pixelů a i, j jsou souřadnice levého horního rohu tohoto bloku v rámci celého snímku. Ve vztahu se také vyskytuje proměnná  $T_c$ , což je korekční matice určená dle HVS.

Dále již postupujeme standardně podle vzorce pro výpočet PSNR (viz 1.2). Výsledná metrika vykazuje podle [5] lepší korelaci se subjektivním měřením, než index MSSIM.

### 1.1.6 PVQM

Protože globální změny obrazu, jako například jas či kontrast, mají za důsledek velmi vysoké chyby při porovnávání jednotlivých pixelů, a zároveň nejsou vnímány jako příliš rušivé, metoda PVQM (Perceptual Video Quality Metric) nejprve sjednotí histogramy referenčního a zkoumaného snímku. [7] Dále také sjednocuje výpočty luminance a chrominance do jednoho, z důvodu mnohem vyšší citlivosti lidského zraku na jas, než na barvu. Signál je dále filtrován podle CSF (funkce kontrastní citlivosti – Contrast Sensitivity Function), která odpovídá lidskému vnímání kontrastu. Následně jsou porovnány jednotlivé pixely referenčního a zkresleného obrazu. Důraz kladen na oblasti hran, kde je zkreslení nejviditelnější. Tato metrika navíc nebere v potaz rychlé výkyvy hodnot v čase, které nejsou pouhým okem pozorovatelné. Výsledné měřítko kvality je bráno v rozsahu od 0 do 85, kde vyšší hodnota představuje lepší kvalitu obrazu.

### 1.1.7 VIF

Metoda VIF (Visual Information Fidelity) je jedna z metod, které vykazují nej-  
přesnější výsledky v porovnání s lidským vnímáním. Její velkou nevýhodou je však  
přílišná komplexnost a tedy i výpočetní složitost.[19] Tato metoda využívá stochas-  
tických modelů pro HVS, zdrojový a zkreslený signál. Základním statistickým mo-  
delem je Gaussovo rozložení a model aditivního šumu. Výpočet se provádí ve frek-  
venční oblasti, do které je signál převeden pomocí vlnkové transformace (DWT –  
Discrete Wavelet Transform). Měřítkem kvality je potom společná informace pro oba  
signály, kterou vypočteme jako množství informace, které lze extrahovat ze zkres-  
leného signálu, v poměru k množství informace signálu původního. Výsledný index  
může teoreticky nabývat hodnot od jedné, pro shodné signály, až po nulu.

Tato metoda lze podle autorů [18] výpočetně zjednodušit tím, že neprovedeme  
převod do frekvenční oblasti, ale počítáme na bázi pixelů. Dalším výkonnostním  
vylepšením je aplikace jednotlivých modelů na bloky o menší velikosti, na rozdíl od  
celého snímku.

### 1.1.8 VSNR

Hlavními prvky metody VSNR (Visual Signal to Noise Ratio) jsou vlnková transfor-  
mace a funkce kontrastní citlivosti CSF dle HVS. [8] Nejprve je vypočtena diference  
původního a zkresleného signálu. Poté je výsledná diference a zkreslený signál trans-  
formován do frekvenční oblasti pomocí DWT. Podle funkce CSF je pro každou frek-  
venci zjištěn práh detekce kontrastu. Dále je zjištěna konkrétní hodnota kontrastu  
zkreslení, opět pro každou frekvenci. Pokud je následně hodnota kontrastu zkreslení  
nižší než práh (není viditelné), VSNR je rovno nekonečnu a měření je ukončeno.  
V opačném případě je kontrast zkreslení přepočten na skutečně vnímaný kontrast  
podle CSF. V tomto bodě mohou být provedeny i některé další korekce. Nakonec je  
vypočtena hodnota VSNR v decibelech:

$$\text{VSNR} = 10 \log_{10} \frac{C^2(\mathbf{I})}{VD^2}, \quad (1.19)$$

kde  $C(\mathbf{I})$  je kontrast zkresleného snímku a  $VD$  (Visual Distortion) je lineární funkce  
kontrastu zkreslení.

Podle autorů a jejich statistických výpočtů v [8], metoda VSNR dosahuje proka-  
zatelně lepších výsledků než PSNR a je srovnatelná s metodou SSIM. Výhodou této  
metody je, že není příliš náročná na operační paměť zařízení provádějícího výpočet.  
Potřebný počet bajtů je obvykle  $2N$ , kde  $N$  je počet pixelů snímku.

## 1.2 Subjektivní metody

Subjektivní metody hodnocení kvality videa spočívají v představení konkrétního snímku velkému počtu pozorovatelů, kteří toto video posuzují. Vybranou statistickou metodou je poté určen výsledný index kvality. Tyto metody lze dále rozdělit podle toho, zda subjektu poskytují pouze jeden podnět (single-stimulus), dva (double-stimulus), nebo více podnětů (multi-stimulus). Při subjektivním hodnocení jednotlivých kodeků hraje důležitou roli také zvolená videosekvence či okolní prostředí, které může ovlivňovat úsudek pozorovatele. Proto byly tyto podmínky v doporučení ITU-T P.910 standardizovány [12].

### 1.2.1 ACR

Při metodě ACR (Absolute Category Rating) jsou jednotlivé výroky pozorovatelů rozděleny dle tabulky 1.1. Ke každé možnosti je také přiřazena číselná hodnota v rozsahu od 1 do 5. Samotné hodnocení probíhá kontinuálně střídáním prezentační a hodnotící periody. Požadovaná sekvence je zobrazována samostatně (single-stimulus) přibližně po dobu 10 s, poté dojde ke ztmavení obrazovky a pozorovatel má čas na udělení subjektivního skóre. Čas pro hodnocení nesmí dle doporučení přesahovat 10 s. [12]

Tab. 1.1: Tabulka kategorií ACR

5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Požadujeme-li přesnější hodnocení pozorovatelů, lze pro tuto metodu použít i devítihodnotovou stupnici, která mezi každé dva již existující stupně přidává jeden celočíselný mezistupeň. Výsledný index kvality určíme jako aritmetický průměr všech pozorování. Tato hodnota se také nazývá MOS (Mean Opinion Score). Vypočtený index kvality MOS by neměl být uváděn bez nejistoty měření, kterou určíme za užití směrodatné odchylky, při spolehlivosti 95 % jako:

$$u_c = 1.96 \frac{s}{\sqrt{N}}, \quad (1.20)$$

kde  $N$  je počet pozorovatelů a  $s$  je výběrová směrodatná odchylka [11]. Směrodatnou odchylku ze souboru jednotlivých udělených skóre můžeme vypočítat klasicky:



$$s = \sqrt{\sum_{i=1}^N \frac{(\bar{v} - v)^2}{(N - 1)}}, \quad (1.21)$$

kde  $N$  je počet pozorovatelů,  $\bar{v}$  je průměrná hodnota uděleného skóre (MOS) a  $v$  je skóre udělené jednotlivými pozorovateli.

### 1.2.2 ACR–HR

Metoda ACR–HR (Absolute Category Rating – Hidden Reference) je velice podobná předchozí metodě s tím rozdílem, že skupina představovaných sekvencí obsahuje navíc referenční video s vynikající kvalitou obrazu. S tímto faktem není pozorovatel seznámen, čímž lze snížit závislost jeho hodnocení na obsahu videa. Od skóre, které bylo uděleno každé videosekvenci, je odečteno skóre její reference a vypočtena hodnota  $DV(PVS)$  (Differential Viewer score per Processed Video Sequence).

$$DV(PVS) = V(PVS) - V(REF) + 5, \quad (1.22)$$

kde  $V(PVS)$  je skóre konkrétního pozorovatele a  $V(REF)$  je skóre udělené referenci tímž pozorovatelem. V případě, že referenční video dostane nižší hodnocení, než video degradované, je i tento výsledek považován za validní [12]. Ze souboru hodnot  $DV(PVS)$  lze vypočítat aritmetickým průměrem výsledný index kvality, který se v tomto případě nazývá DMOS (Differential Mean Opinion Score).

### 1.2.3 DCR

Testování metodou DCR (Degradation Category Rating) zahrnuje jak referenční, tak zpracovanou sekvenci. Nejdříve je vždy představena sekvence referenční, poté je obrazovka ztmavena na 2 s, a až posléze zobrazena sekvence testovací. (double–stimulus) Následuje čas na hodnocení, který by opět neměl přesáhnout 10 s. Pozorovatel hodnotí míru degradace v porovnání s referencí a má možnost zvolit mezi pěti kategoriemi (viz tab. 1.2). Výsledné MOS skóre dostaneme stejným způsobem jako u metody ACR.

Tab. 1.2: Tabulka kategorií DCR

5	Imperceptible
4	Perceptible but not annoying
3	Slightly annoying
2	Annoying
1	Very Annoying

### 1.2.4 PC

Během metody PC (Pair Comparison) jsou z testovacího souboru prezentovány současně vždy dvě videosekvence takovým způsobem, aby za dobu testování byly zobrazeny všechny možné kombinace dvojic, u kterých záleží na pořadí. Úkolem subjektu je pouze zvolit tu sekvenci, kterou považuje za obrazově kvalitnější. Prezentační a hodnotící časy se oproti předchozím metodám nemění. Výstupem této metody je pořadí jednotlivých kodeků, reprezentující jejich výslednou kvalitu videa.

### 1.2.5 CCR

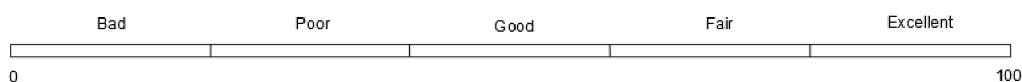
Metoda CCR (Comparison Category Rating) je opět metodou využívající dva podněty (double-stimulus). Zobrazování snímků probíhá stejně jako při metodě DCR, avšak referenční a testovací snímky jsou zobrazovány v náhodném pořadí tak, aby pozorovatel nevěděl, který snímek je referenční. Kategorická stupnice nabývá hodnot od  $-3$  po  $3$ , kde pozorovatel vyjadřuje kvalitu obrazu druhé sekvence (viz tab. 1.3). Tato metoda dobře minimalizuje vliv obsahu videosekvence na hodnocení pozorovatele, avšak je velice náchylná k neúmyslné záměně kladých hodnot za záporné a naopak. [1]

Tab. 1.3: Tabulka kategorií CCR

$-3$	Much worse
$-2$	Worse
$-1$	Slightly worse
$0$	The same
$1$	Slightly better
$2$	Better
$3$	Much better

### 1.2.6 SSCQE

Kvalita kódovaného videa není vždy konstantní v čase, naopak může kolísat jak při statické, tak i při dynamicky se měnící scéně. Předchozí metody navíc nedokáží spolehlivě zachytit krátkodobé distorze, které mohou při kódování videa vznikat. Z tohoto důvodu byla uvedena metoda SSCQE (Single Stimulus Continuous Quality Evaluation) [11], při které je současně zobrazováno pouze jedno video, avšak delšího průběhu. Hodnotící škála je spojitá v rozsahu od 0 do 100 a opatřena již známými ACR kategoriemi Bad – Excelent, které jsou na této stupnici lineárně rozmístěny (viz obr. 1.2).



Obr. 1.2: Škála pro hodnocení metodou SSCQE [11]

Úkolem pozorovatele je v reálném čase měnit pozici posuvníku dle aktuálně vnímané obrazové kvality. Výstupem každého pozorování je závislost indexu kvality na čase. Všechny funkce jsou poté zprůměrovány do výsledného průběhu  $q(t)$ . Menší nevýhodou této metody může být časová prodleva mezi okamžikem poklesu kvality a skutečnou reakcí pozorovatele, který dostane posuvník do odpovídající polohy.

### 1.2.7 DSCQS

Při metodě DSCQS (Double Stimulus Continuous Quality Scale) je prezentován pár referenčního a komprimovaného videa v náhodném pořadí. Tuto metodu lze podle [11] provést dvěma způsoby. První varianta dovoluje pozorovateli několikrát přepnout mezi dvěma snímky, které běží po dobu až 10 s, a tímto způsobem si důkladně rozmyslet výsledné hodnocení. Při druhé variantě hodnotí prezentované sekvence více osob najednou. Pár snímků je zobrazen opakovaně, teprve při posledním zobrazení probíhá záznam všech hodnocení. Počet opakování závisí na délce sekvence. Pozorovatel hodnotí celkovou kvalitu videa na spojitě stupnici (viz obr. 1.2) s tím rozdílem, že má k dispozici měřítko pro obě videosekvence. Při analýze výsledků je nejprve určen rozdíl mezi referenční a testovací sekvencí a poté vypočtena střední hodnota pro všechny pozorovatele.

## 2 ZKOUMANÉ VIDEO KODEKY

### 2.1 H.264

Kodek H.264 podle standardu H.264/AVC (Advanced Video Coding) je v současné době nejpoužívanější ze všech video kodeků. Lze ho nalézt na většině videoportálů, Blu-ray discích či téměř ve všech typech HDTV vysílání. Uvedený standard nezahrnuje postup kódování videa, zde záleží na konkrétní implementaci výrobce kodéru, ale obsahuje pouze formát a syntaxi zakódovaného videa a postup jeho dekódování [17].

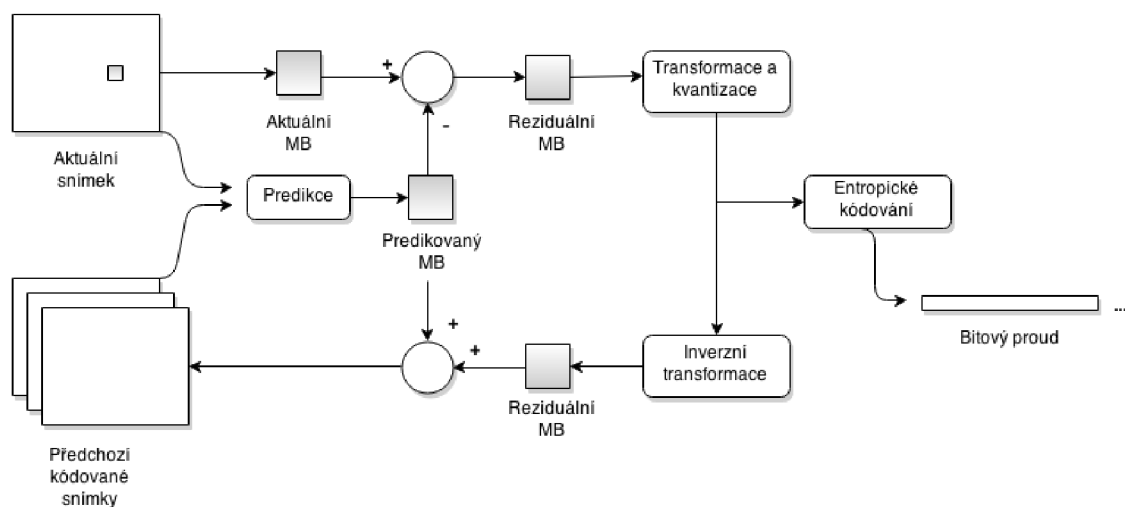
Kódování videa probíhá typicky užitím makrobloků o velikosti 16 x 16 pixelů. Jednotlivé makrobloky jsou sdružovány do snímků. Snímky I (Intra) jsou referenční a mohou být kódovány samostatně. P (Prediction) snímky dostaneme predikcí z předchozího referenčního snímku. Snímky typu B (Bidirectional) lze vytvořit jak ze snímků I, tak i ze snímků P a to v libovolném směru. Kromě klasických snímků I, P, B byly zavedeny také snímky SI a SP (Switching-I a Switching-P), které slouží pro snazší přepínání mezi bitovými proudy, například při streamování videa v několika odlišných kvalitách současně.

#### Kódování a dekódování

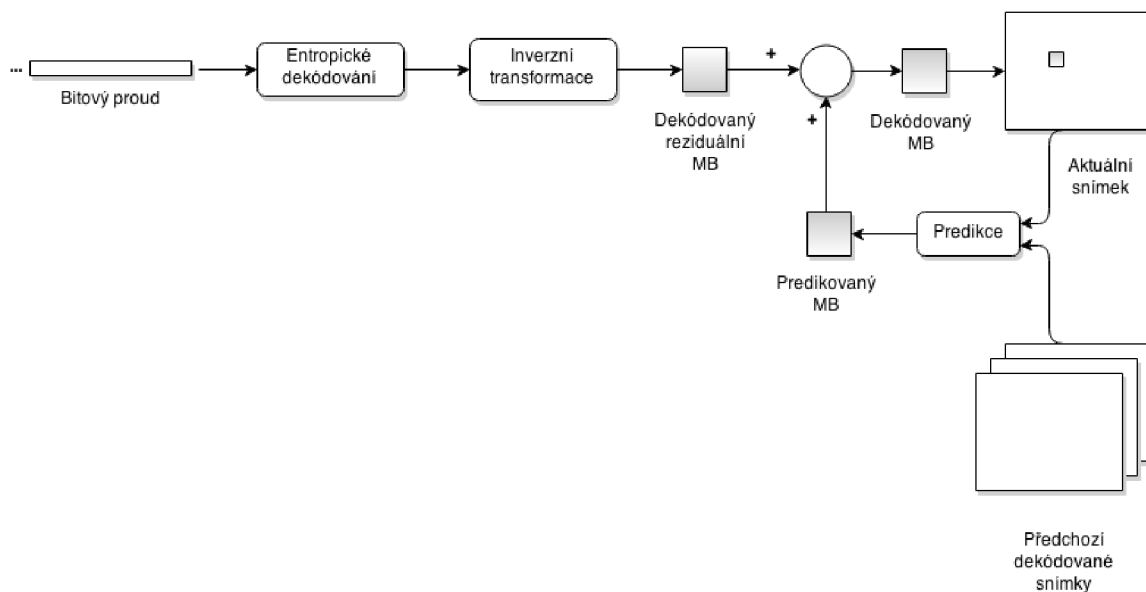
Při kódování jsou vytvářeny predikované makrobloky na základě předchozích kódovaných snímků (intrapredikce), nebo na základě okolních, již zakódovaných pixelů, v rámci jednoho snímku (interpredikce). (obr. 2.1) Predikce makrobloků může využívat různé techniky odhadu pohybu, zejména pak pohybové vektory. Pokud je aktuálně kódovaný makroblok nalezen v blízkém okolí v jeho referenčním snímku, je vypočten pohybový vektor, který udává přesnou pozici makrobloku. Dále se kódují pouze souřadnice tohoto vektoru, čímž lze ve výsledku ušetřit velké množství bitů. Souřadnice tohoto vektoru mají u kodeku H.264 přesnost až 1/4 pixelu.

Predikovaný makroblok je poté odečten od aktuálně zpracovávaného makrobloku. Tento proces se nazývá kompenzace pohybu. Výsledný reziduální makroblok je transformován užitím celočíselné diskretní kosinové transformace (DCT). Jednotlivé koeficienty jsou kvantizovány dle daného kvantizačního kroku. Posledním krokem k tomu, abychom dostali výsledný bitový proud, je entropické kódování. Lze použít aritmetické kódování, či kódování s proměnnou délkou kódu VLC (Variable Length Coding). Do výsledného bitového proudu jsou také zahrnuty všechny další informace o videu dle stanovené syntaxe. Ve zpětné větvi kodéru probíhá zároveň také inverzní transformace. Výsledný reziduální makroblok je přičten k predikovanému a uchován v paměti pro pozdější predikce.

Při dekódování (obr. 2.2) je získán inverzními operacemi entropického kódování a transformace dekódovaný reziduální makroblok. Z již dekódovaných snímků je opět stejným způsobem jako při kódování vytvořen predikovaný makroblok. Výsledný dekódovaný makroblok dostaneme jako součet predikovaného a reziduálního makrobloku. Po sestavení jednotlivých snímků jsou na sekvenci aplikovány filtry pro zlepšení kvality obrazu, zejména filtr odstraňující blokové artefakty (Deblocking Filter).



Obr. 2.1: Obecný postup kódování H.264. [17]



Obr. 2.2: Obecný postup dekódování H.264. [17]

## Profily

Součástí standardu H.264 jsou také nástroje a postupy pro kódování videa. Konkrétní sada definovaných postupů při kódování videa se nazývá profil. Profily mají za účel optimalizovat videosekvenci pro konkrétní požadavky různých zařízení či aplikací. [17]

Základní profil (Baseline) byl určen zejména pro přenos videa v reálném čase, například videokonference a méně výkonná zařízení, jako například mobilní telefony. Obsahuje pouze I a P snímky. B snímky jsou pro snížení výpočetní složitosti zcela vypuštěny. Tento profil používá entropické kódování CAVLC (Context Adaptive Variable Length Coding).

Rozšířený profil (Extended) je velice podobný základnímu s tím rozdílem, že obsahuje vylepšení pro dokonalejší síťový přenos a streamování videa, například SI a SP snímky pro snadný přechod mezi bitovými rychlostmi.

Hlavní profil (Main) je určený pro televizní DVB vysílání v standardní kvalitě nebo DVD přehrávače. Entropické kódování je na rozdíl od předchozích profilů použito typu CABAC (Context Adaptive Binary Arithmetic Coding). Při kódování jsou již také zahrnuty B snímky.

Vysoký profil (High) klade důraz na kvalitu výsledného obrazu, za cenu vyšší hardwarové náročnosti dekodéru či podstatně delší doby kódování. Tento profil má ještě další rozšíření, která zvyšují například bitovou hloubku na 10 bitů nebo vzorkování videa YCbCr z původního 4:2:0 na 4:2:2 či dokonce 4:4:4.

Spolu s profily jsou definovány také takzvané úrovně. Ty určují parametry videa, které je dekodér s určitou úrovní ještě schopen zpracovat. Jednotlivé úrovně jsou číslovány od 1 do 5 s tím, že tyto úrovně mají ještě další mezikroky. Parametry definované těmito úrovněmi jsou například maximální bitová rychlost, maximální velikost snímku v makroblocích či počet makrobloků za sekundu. Nejvyšší podporované rozlišení kodeku H.264 je 4K UHD, tedy 4096 x 2304 pixelů.

## 2.2 H.265

Kodek H.265/HEVC (High Efficiency Video Coding) je nejnovějším standardizovaným kompresním formátem. Je orientován především na videa ve velmi vysoké kvalitě. V současné době není příliš rozšířený, ale má potenciál v budoucnu postupně kompletně nahradit kodek H.264. Obecné postupy kódování a dekódování videa se v principu nijak neliší od předchozího standardu, avšak na rozdíl od H.264 slibuje tento kodek mnohem vyšší kompresi při zachování obrazové kvality, a to především díky lepším technikám predikce. Úspora bitové rychlosti má být dle některých testů až 50 %. [22]

Velikost kódovaných makrobloků může být nově i 32 x 32 či dokonce 64 x 64 pixelů, což napomáhá rychlosti komprese. Dalším výkonnostním vylepšením je rozdělení jednotlivých snímků na takzvané dlaždice, které mohou být nezávisle zpracovávány jednotlivými vlákny procesu, čímž odpadá jejich nadbytečná synchronizace. Paralelizaci napomáhá také technika WPP (Wavefront Parallel Processing), která umožňuje více vláknům efektivně zpracovávat snímek po jednotlivých řádcích tvořených strukturami makrobloků připravených ke zpracování, tzv. CTU (Code Tree Unit). Tato technika zároveň pracuje v souladu s činností entropického kodéru. [9].

DCT lze nově provést i na jednotlivých blocích o velikosti až 32 x 32 pixelů, na rozdíl od předchozí maximální velikosti 8 x 8, a zároveň tyto bloky nemusí být čtvercové. Pro některé typy intrapredikce byla dokonce zahrnuta i celočíselná diskrétní sinová transformace (DST) [16]. Zásobník dekódovaných snímků DPB (Decoded Picture Buffer), které lze použít pro predikci, byl rozšířen z maximálních šesti snímků u kodeku H.264 až na 16 snímků, v závislosti na použitém profilu a úrovni.

### Profily

Kodek H.265 obsahuje, kromě několika rozšiřujících profilů, které jsou určeny pro zvláštní účely jako například kódování statických obrazů nebo monochromatické snímky, pouze hlavní profil (Main). V něm je definováno vzorkování YCbCr 4:2:0 a bitová hloubka obrazu 8 bitů. Tento profil má však několik odlišných verzí, které mají v názvu vždy vzorkování a bitovou hloubku. Další možné bitové hloubky jsou

10 a 12 bitů a vzorkování 4:2:2 nebo 4:4:4. Maximální podporovaná bitová hloubka je 16 bitů při vzorkování 4:4:4 a použití pouze I snímků a to jen v nejvyšším profilu Main 4:4:4 16 Intra.

Standardem definované úrovně jsou v rozsahu 1 až 6, přičemž nejvyšší úroveň má hodnotu 6.2. Celá šestá úroveň podporuje rozlišení 8K, tedy 8192 x 4320 pixelů. Pro lepší cílení na jednotlivá zařízení definuje standard kromě úrovní ještě dva stupně, hlavní (Main) a vysoký (High), které se liší maximální bitovou rychlostí.

## 2.3 VP8

Zatímco předchozí kodeky podléhají různým patentům, VP8 je kodek, který byl vyvíjen firmou Google pod licencí CC (Creative Commons), tedy zdarma k užití. Z patentových důvodů také tento kodek nepoužívá predikované B snímky. Obecný proces kódování a dekódování odpovídá kodeku H.264, avšak jednotlivé konkrétní postupy se liší. Namísto B snímků používá kodek střídavé referenční snímky, které mohou ale nemusí být viditelné. Ty je možné vytvořit z libovolného počtu již zakódovaných i nezakódovaných snímků a zároveň mohou sloužit jako reference pro libovolný počet již zakódovaných P snímků. Touto technikou lze podstatně zvýšit rychlost dekódování, která je podle [4] asi o 30 % vyšší než u H.264. Mimo tyto snímky používá kodek i takzvané zlaté snímky. Zlatý snímek je snímek, který je vhodný jako referenční při snímkové predikci. Zlatým snímkem může být jakýkoli P snímek a zároveň jím jsou všechny I snímky.

Pro intrapredikci definuje VP8 čtyři hlavní módy, H\_PRED (Horizontal Prediction), V\_PRED (Vertical Prediction), DC\_PRED (DC Prediction) a TM\_PRED (True Motion Prediction), které se liší zejména oblastí určenou k predikci, dále pak několik rozšiřujících módů. Interpredikční bloky jsou konstruovány převážně pomocí pohybových vektorů, které směřují na daný referenční blok. Kodek VP8 při kódování pohybových vektorů využívá i již zakódovaných vektorů z okolních bloků, a zároveň dokáže efektivně rozdělit makroblok na různě velké části a tím vylepšit proces interpedikce.

Kodek VP8 transformuje bloky o velikosti 4 x 4 pixely pomocí DCT, která není celočíselná, a na rozdíl od H.264 také pomocí WHT (Walsh-Hadamard Transform). [2] V kodéru je poté použito aritmetické kódování odvozené od Huffmanova, výsledný bitový proud je rozdělen na více oddílů, které mohou být lépe zpracovány více procesy či vlákny, čímž se zvyšuje rychlost kódování a dekódování. Specifický pro tento kodek je také způsob použití filtrů, které odstraňují například blokové artefakty. Ty jsou aplikovány pouze na určité části snímku v různé míře, především jsou vynechány části se statickým obsahem. Na rozdíl od H.264 tento kodek neob-



sahuje žádné profily či úrovně. Vzorkování je použito standardně 4:2:0. Maximální podporované rozlišení je 16348 x 16384 pixelů, což ale v praxi nemá využití.

## 2.4 VP9

Kodek VP9, nástupce kodeku VP8, je podobně jako H.265 zaměřen na video ve velmi vysoké kvalitě. Pro vyšší efektivitu kódování bylo nutné, podobně jako u kodeku H.265, navýšit velikost blokových oblastí až na 64 x 64 pixelů. Tyto bloky se nazývají superbloky (SB). Ty mohou být dále děleny na menší superbloky 32 x 32 a makrobloky 16 x 16 pixelů. Superbloky mohou být také predikovány jako jeden celek, podobně jako makrobloky. Makrobloky mohou být dále děleny pro jemnější predikci a účely transformace až do velikosti 4 x 4 pixelů. [6] VP9 podporuje deset intrapredikčních módů, ve kterých jsou definovány predikce i v diagonálním směru. Interpredikce může být realizována až ze čtyř referenčních snímků, které mohou být vybrány až z osmi možných.

Pro různé módy predikce jsou použity různé transformace. DCT pro bloky o velikosti 4 x 4 až 32 x 32 pixelů při intrapredikci, hybridní DCT/ADST (Asymmetric Discrete Cosine Transform) pro interpredikci. Pouze pro bezztrátové kódování je použita transformace WHT. Velikost bloku pro transformaci je obvykle volena jako maximální možná, tedy rovna velikosti predikčního bloku. Entropické kódování je zde použito výhradně aritmetické s binárním stromem.

Pro zpracování videa ve vysoké kvalitě je nutné efektivně využít paralelního zpracování procesů. Kodek VP9 proto podporuje podobně jako H.265 rozdělení jednotlivých snímků na dlaždice, zároveň také v hlavičkách jednotlivých rámců bitového proudu uvádí značky, které umožňují synchronizaci paralelního dekódování tohoto proudu.

Ve srovnání s předchozím kodekem, VP9 nově zavádí dva profily 0 a profil 1. Profil 0 je základní profil se vzorkováním YCbCr 4:2:0 při bitové hloubce 8 bitů, zatímco profil 1 rozšiřuje možnosti vzorkování i na 4:2:2 a 4:4:4. [14]

## 2.5 Dirac

Dirac je otevřený, volně šiřitelný kodek vyvinutý společností BBC. Tento kodek byl standardizován organizací SMPTE (Society of Motion Picture and Television Engineers) pod názvem VC-2. Jeho poslední verze nese označení 2.2.3. Na rozdíl od všech uvedených kodeků, Dirac spoléhá pouze na použití diskrétní vlnkové transformace DWT. Ta je dostatečně flexibilní a umožňuje použít celkově menší počet základních nástrojů pro kódování, což činí tento kodek velice kompaktní. Kódovat lze všechna

dostupná rozlišení včetně 8K UHD při vzorkování 4:2:0, 4:2:2 či 4:4:4. Maximální dostupná bitová hloubka je 16 bitů. [21]

Při kódování využívá kodek Dirac vlastní techniky kompenzace pohybu nazvané OBMC (Overlapped Block Motion Compensation), která redukuje výskyt blokových artefaktů v kódovaném snímku při větších velikostech bloků. Predikce snímků je zde implementována za použití vektorů pohybu, které mohou mít přesnost až 1/8 pixelu. Dirac během kódování vytváří jak I, tak i P a B snímky. Výsledný bitový proud je zakódován pomocí entropického kodéru.

Ve své specifikaci Dirac uvádí čtyři profily. Main (Long GOP), Main (Intra), Simple a Low Delay. Profil Main (Long GOP) má obecně nejlepší poměr kvality a bitové rychlosti. Využívá delší sekvence predikovaných snímků tzv. GOP (Group Of Pictures) a aritmetického kódování. Profil Main (Intra) používá také aritmetické kódování, avšak pouze snímky I, což vede k rychlejšímu kódování a dekodování za cenu vyšší bitové rychlosti, při zachování stejné obrazové kvality. Profil Simple navíc nepoužívá aritmetické kódování, čímž se proces kódování a dekodování ještě zjednoduší. Profil Low Delay je výsledkem optimalizace pro streamování videa přes internet.

Dirac zavádí také tři úrovně, definující minimální požadavky na dekodér. Úroveň 0 je rezervována pro videosekvence, které nevyhovují následujícím dvěma úrovním. Úroveň 1 lze použít pouze pro profily Main (Intra), Simple a Low Delay. Poslední úroveň nese označení 128 a je určena pouze pro použití s profilem Main (GOP).

## 2.6 WMV

Kodek WMV (Windows Media Video) byl v jeho nejnovější verzi 9 standardizován organizací SMPTE pod označením VC-1 [20]. Při kódování využívá obecně známých technik kompenzace a odhadu pohybu a vytváří I, P i B snímky. Kodek VC-1 zjednodušuje proces odhadu pohybu omezením referenčních snímků, ze kterých může být P snímek vytvořen pouze na dva. B snímek je tvořen vždy čtyřmi referenčními snímky, které mohou být typu I nebo P. Velikosti kódovaných bloků jsou také omezeny pouze na dvě a to 16 x 16 nebo 8 x 8. V případě že je kódovaný snímek shodný s referenčním, není kódován vůbec, ale je vynechán. Při dekodování je poté na jeho místě dosazen snímek referenční. Kodek nemá stanovenou délku sekvence predikovaných snímků GOP, struktura a počet I, P, B snímků se může lišit [13].

Vstupním formátem může být pouze YUV 4:2:0, z čehož vyplývá že kodek podporuje pouze vzorkování YCbCr 4:2:0. Použitá transformace DCT je celočíselná s variabilní velikostí bloků. Lze použít velikost 4 x 4, 8 x 8 či kombinace těchto rozměrů. Pro snížení výskytu blokových artefaktů a vylepšení obrazové kvality jsou použity

dvě specifické techniky OLT (Overlapped Transform) a ILF (In-loop Deblocking Filter). Při technice OLT dochází během inverzní transformace k difuzi sousedních bloků a tím snížení výskytu viditelných hran. ILF aplikuje na oblasti okolo hran jednotlivých bloků nelineární filtr typu dolní propust. Tato filtrace je provedena pouze u snímků I a P, avšak díky referencím převezmou provedené změny i B snímky. Výstupní bitový proud je kódován pomocí VLC kodéru.

VC-1 definuje tři profily. Simple, který je určen pro videa s velice nízkým rozlišením. Main, který při úrovni High může kódovat i Full HD video při maximální bitové rychlosti 20 Mb/s. Advanced, při kterém lze kódovat video s maximálním rozlišením 2048 x 1536 pixelů při běžné snímkové rychlosti 24 fps.

## 2.7 Sorenson Spark

Kodek Sorenson Spark, často označovaný jako Flash Video (Flash Video je pouze kontejner), vychází z velké části z proprietárního kodeku H.263. Některé vlastnosti kodeku H.263 byly při implementaci odstraněny a naopak jiné nově doplněny. Odstraněny byly zejména takzvané PB snímky, které navzájem prokládají makrobloky z P a B snímků. Dále pak skupiny bloků GOB (Group of Blocks), které sdružují makrobloky z jednoho snímku a přidávají jim vlastní hlavičku. Odstraněna byla také možnost použití kódování SAC (Syntax-based Arithmetic coding) a namísto něj použito pouze kódování s proměnnou délkou kódu VLC. Doplněna byla možnost kódování videa s libovolnou šířkou a výškou do maximálního rozměru 65535 pixelů [23].

Kódované bloky mohou mít rozměr od 16 po 256 pixelů, které nemusí být čtvercové, avšak jednotlivé rozměry musí být násobkem šestnácti. V případě že rozměr snímku není celočíselným násobkem šestnácti, okrajové bloky jsou pouze částečné a obsahují jen pixely nutné k doplnění tohoto rozměru. Aby mohly být uvedené změny implementovány, bylo nutné změnit hlavičku, kodek Sorenson proto uvádí svou vlastní syntaxi, odlišnou od H.263.

### 3 DRUHY CHYB

Při streamování videa přes internet se na přenosové cestě může vyskytnout celá řada okolností, které mohou různými způsoby poškodit přenášená data. Nejčastější příčinou jsou interference s jinými zařízeními, které mohou vzniknout překryvem kanálů při bezdrátovém přenosu. Interference mohou obecně způsobovat jakákoli zařízení vyzařující elektromagnetické záření. U kabelového přenosu mohou být příčinou zejména přeslechy (tzv. crosstalk). Dalším zdrojem chyb může být útlum při přenosu, který způsobí snížení úrovně přijímaného signálů. Ten je při regeneraci zesílen i s přítomným šumem, který je poté zdrojem chyb. Chyby mohou vznikat také nesprávnou synchronizací přijímače a vysílače. U streamování videa může také dojít k přetížení sítě či serveru, který nestíhá obsloužit všechny požadavky a dochází buďto k úplným výpadkům přenosu či zahazování jednotlivých paketů. [24]

Nejčastějším druhem chyb jsou chyby náhodné, kdy dochází k nepravdělným záměnám jednotlivých bitů. Tyto chyby jsou často dobře opravitelné kanálovým kódováním, tzv. samoopravnými kódy. Zdrojem těchto chyb je obvykle vyšší úroveň šumu na pozadí. V obrazovém signálu se obvykle projevují jako bodové či blokové barevné odlišnosti.

Dalším druhem chyb jsou takzvané bursty, neboli shluky chyb. Ty jsou často důsledkem interferencí s jinými signály. Základním parametrem burstu je jeho délka, která udává vzdálenost od prvního chybně přijatého bitu po poslední chybně přijatý bit za předpokladu, že v této sekvenci nebyl přijat žádný správný symbol. Základním opatřením pro ochranu dat proti burstům je tzv. prokládání (Interleaving), které jednotlivé bursty rozdělí na snáze opravitelné náhodné chyby. I proti burstům se lze bránit kanálovým kódováním. Velice účinné jsou, dnes často používané, Reed–Solomonovy kódy. [24]

Nejhorším případem chyb jsou úplné výpadky signálu, často způsobené zahazováním paketů. Tomuto druhu chyb je někdy obtížné předcházet.

Důležitá veličina, která popisuje chybovost kanálu se nazývá bitová chybovost neboli BER (Bit Error Rate) a je definována jako poměr chybně přijatých bitů k celkovému počtu přijatých bitů. Tato veličina hraje důležitou roli při optimalizaci jednotlivých parametrů přenosových kanálů, jako je například šířka pásma nebo vysílací výkon [3].

Kromě BER definujeme také chybovost symbolovou neboli SER (Symbol Error Rate), která udává počet chybně přijatých symbolů k celkovému počtu přijatých symbolů. Typickým symbolem je ve většině případů 1 bajt.

## 4 PRAKTICKÁ ČÁST

V následující části bude podrobně popsána aplikace VideoCodecs, která je cílem této práce. Finální verze v podobě desktopové aplikace obsahuje zejména tyto čtyři hlavní funkce pro usnadnění měření a práce s videem:

- kódování videa pomocí všech výše uvedených video kodeků,
- dekódování videa,
- objektivní hodnocení kvality obrazu užitím algoritmů PSNR, SSIM a M-SVD,
- simulaci přenosového kanálu zanesením chyb do vloženého souboru.

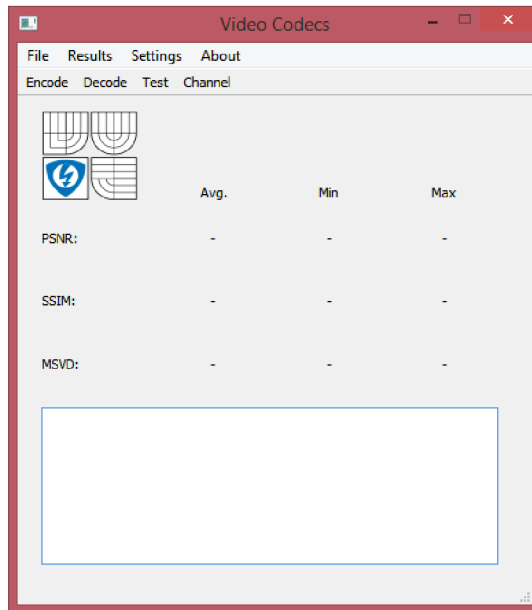
Za pomoci této aplikace bylo provedeno srovnání obrazové kvality jednotlivých video kodeků. Průběh a kompletní výsledky provedeného měření jsou popsány v sekcích 4.2 a 4.3.

### 4.1 Realizace

Aplikace VideoCodecs byla vytvořena užitím programovacího jazyku C++. Pro realizaci grafického uživatelského rozhraní GUI (Graphical User Interface) byl použit aplikační framework Qt 5.4.1 a vývojové prostředí Qt Creator ve verzi 3.4.0. Pro kódování a dekódování videa je použita knihovna ffmpeg sestavení 2.6.1. Další nezbytnou externí knihovnou je dynamická knihovna `opencv_core2410.dll`, která je použita pro rozklad matic na jejich singulární hodnoty (viz 1.1.3).

Hlavní okno aplikace (obr. 4.1) obsahuje tři uživatelské prvky. V horní části lištu s rychlým přístupem k jednotlivým akcím, ty budu popsány níže. Tabulku pro náhled výsledků konkrétního měření a ve spodní části textovou oblast, sloužící pro výpis činností programu a dalších informací.

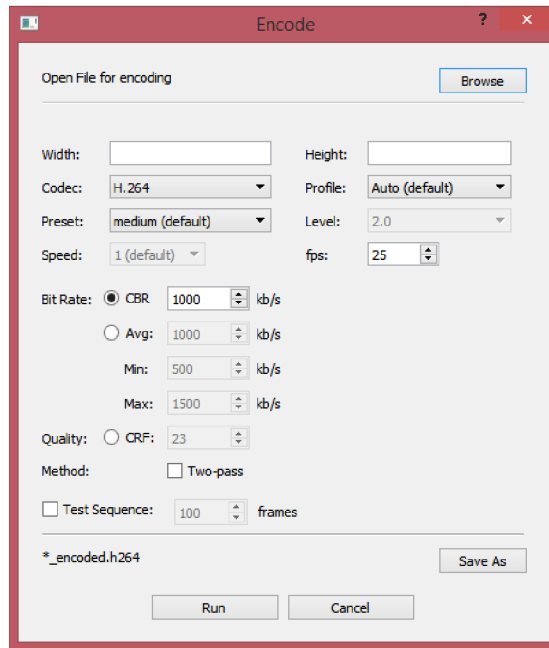
Pro kódování videa slouží okno *Encode* (obr. 4.2). Vstupním formátem je soubor `.yuv`, který obsahuje nezakódované video se vzorkováním YUV 4:2:0. Jelikož tento soubor neobsahuje žádnou hlavičku, ale pouze syrová data, je nutné zadat rozměry videa v pixelech a jeho snímkovou rychlost *fps*. V položce *Codec* lze vybrat ze všech posledních verzí kodeků popsaných v této práci, s výjimkou kodeku WMV9. Tento kodek je knihovnou ffmpeg podporován pouze ve verzi 8, proto je i tato verze použita pro testování v další kapitole. Ostatní parametry kódování lze volit v závislosti na zvoleném kodeku. U kodeků H.26x je to zejména položka *preset*, kterou lze volit požadovanou rychlost kódování. Parametr *preset* automaticky řídí nastavení profilu a úrovně dle vstupního videa a zvolené hodnoty. Platí pravidlo, že čím vyšší rychlost kódování, tím je výstupní kvalita videa horší. U kodeku H.264 lze profil a úroveň nastavit manuálně, u kodeku H.265 se tento postup nedoporučuje. Kodek VP9 má jako alternativu presetů parametr *speed*, který pracuje na stejném principu. U ostatních



Obr. 4.1: Hlavní okno programu VideoCodecs.

kodeků jsou tyto parametry v aplikaci voleny automaticky, nebo nejsou v knihovně ffmpeg implementovány.

V další sekci okna *Encode* lze zvolit mezi konstantní bitovou rychlostí CBR (Constant Bit Rate), proměnnou bitovou rychlostí VBR (Variable Bit Rate) či kvalitu výstupního videa zvolit pomocí faktoru CRF (Constant Rate Factor) u kodeků, které ho podporují. Faktor CRF udržuje konstantní hodnotu kvantizačního parametru a nabývá hodnot od 0 do 51, kde 0 představuje bezztrátové kódování a 51 nejhorší kvalitu [10]. Zvolit lze také dva průchody kodérem, což zaručí přesnější bitovou rychlost na výstupu. Poslední možností je *Test sequence*, kde lze zadat počet snímků ke kódování. Tuto možnost je dobré využít k testování zvolených parametrů kodéru, kde není žádoucí čekat na kódování celé sekvence.



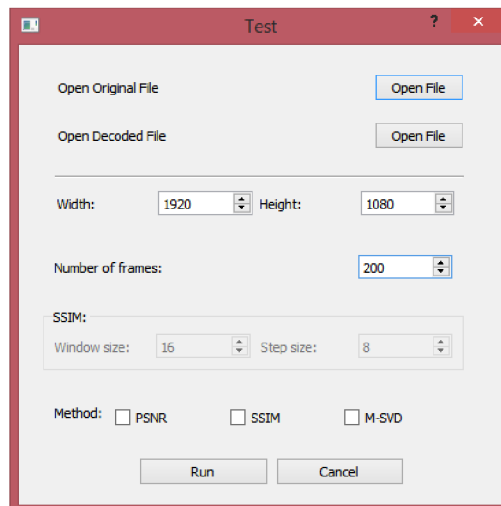
Obr. 4.2: Okno *Encode* pro kódování videa do požadovaných formátů.

Okno *Decode* nabízí pouze dvě volby a to vstupní video soubor a umístění dekódovaného souboru. Dekódovaný soubor je opět uložen ve formátu .yuv a progresivním vzorkování YUV 4:2:0.

Okno *Test* (obr. 4.3) nabízí možnost otestovat obrazovou kvalitu jednotlivých kodeků. Pro tyto účely byly implementovány objektivní metody PSNR, SSIM a M-SVD. Originální a dekódovaný soubor musí být opět ve formátu YUV 4:2:0. V položce *Number of frames* zvolíme počet snímků, pro které bude měření provedeno. U metody SSIM je třeba zvolit velikost okna a krok, kterým budou jednotlivé snímky procházeny. U metody M-SVD je velikost tohoto okna fixních 8 x 8, z důvodu nepřiměřené náročnosti na výpočet singulárního rozkladu při vyšších řádech matice. V dolní části okna pak můžeme vybrat metody, kterými budou zadané sekvence testovány.

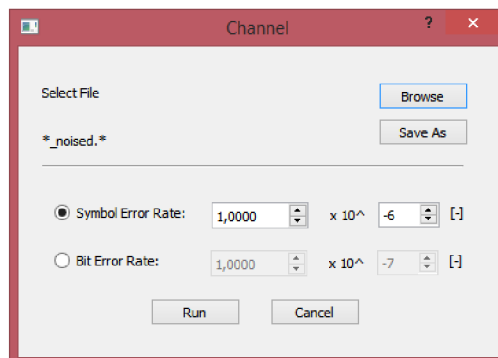
Testování spustíme tlačítkem *Run*. Tlačítkem *Cancel* je možné průběh testování přerušit. Informace o průběhu testování jednotlivými metodami se zobrazují v textové části hlavního okně. Po dokončení jednotlivých metod se zobrazují jejich souhrnné výsledky v tabulce v hlavního okně. Po ukončení výpočtu je možné v hlavního okně v menu *Results* exportovat výsledky do textového souboru CSV (Comma Separated Values), nebo je zobrazit graficky v menu *Results -> Show Graph*.

Poslední funkcí této aplikace je okno *Channel* (obr. 4.4), které slouží pro simulaci přenosového kanálu, zanesením náhodných chyb do vloženého souboru. Tuto funkci lze využít například pro testování závislosti obrazové kvality na chybovosti



Obr. 4.3: Okno *Test* pro měření kvality videa.

kanálu. Volit lze mezi zadáním symbolové chybovosti SER (Symbol Error Rate), v tom případě se spustí algoritmus implementovaný v knihovně ffmpeg, nebo bitové chybovosti BER (Bit Error Rate), jež spustí algoritmus napsaný pro tuto aplikaci.

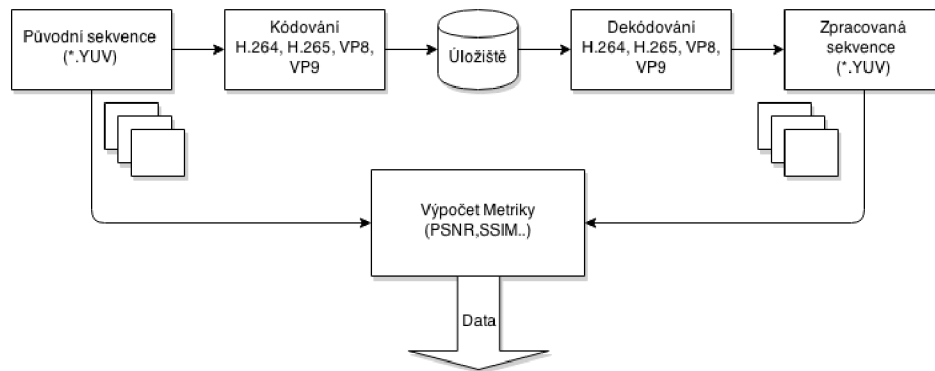


Obr. 4.4: Okno *Channel* sloužící pro simulaci přenosového kanálu.

## 4.2 Metody

Obecný postup testování je naznačen na obr. 4.5. Referenční video ve formátu YUV je zakódováno vybranými kodeky, čímž dojde k jeho degradaci, a poté zpětně dekodováno. Původní a dekodovaný soubor byly následně porovnávány užitím algoritmů PSNR, SSIM a M-SVD, které jsou součástí aplikace VideoCodecs.





Obr. 4.5: Obecný postup hodnocení kvality videa plně referenční metodou.

Pro srovnání všech prezentovaných kodeků byly zvoleny čtyři videosekvence zachycující odlišné typy scén. Mobcal v HD rozlišení 1280 x 720 pixelů při 50 fps (obr. 4.6), Blue sky v rozlišení Full HD 1920 x 1080 pixelů při snímkové rychlosti 25 fps (obr. 4.7), Life ve stejném rozlišení a snímkové rychlosti 30 fps (obr. 4.8) a Old town opět ve Full HD rozlišení při snímkové rychlosti 50 fps (obr. 4.9) [32]. Tyto a další volně šiřitelné sekvence, sloužící pro testovací účely, lze získat na internetové adrese <<https://media.xiph.org/video/derf/>>. Uvedená videa byla zakódována při čtyřech konstantních bitových rychlostech 2500, 5000, 7500 a 10000 kb/s. Použito bylo také dvojího průchodu kóděrem pro přesnější výstupní bitovou rychlost. U všech kodeků bylo ponecháno výchozí nastavení všech ostatních parametrů.



Obr. 4.6: Ukázkový snímek z videa Mobcal



Obr. 4.7: Ukázkový snímek z videa Blue sky



Obr. 4.8: Ukázkový snímek z videa Life



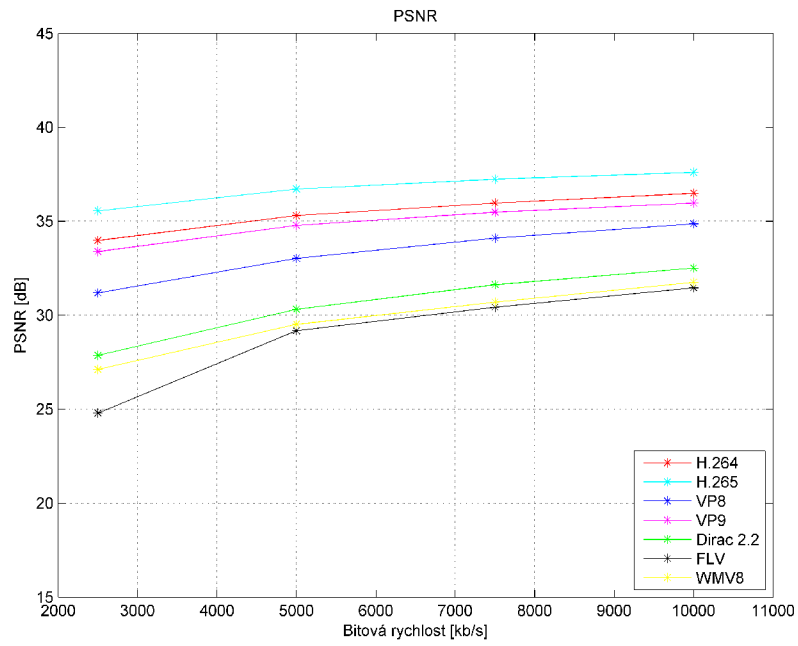
Obr. 4.9: Ukázkový snímek z videa Old town

## 4.3 Výsledky

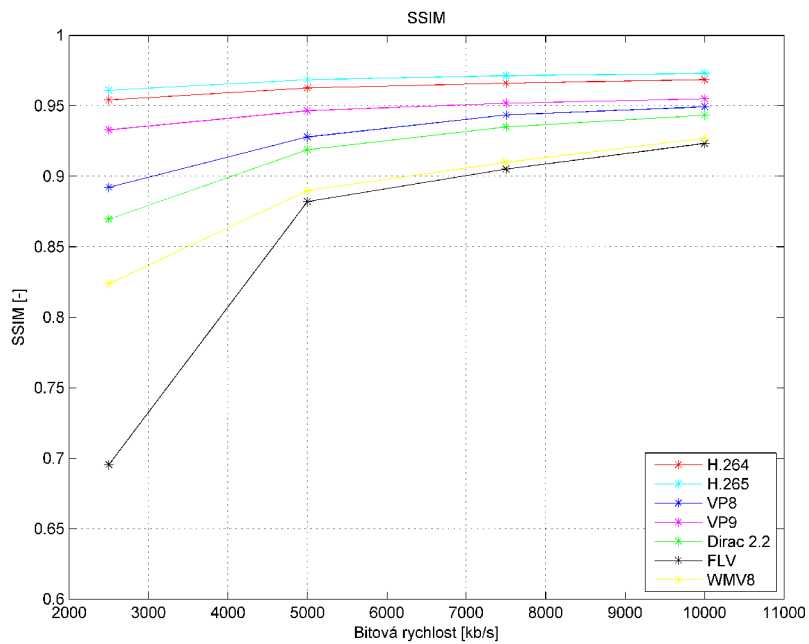
Veškerých výsledků měření bylo dosaženo na PC s procesorem Intel(R) Core(TM) i5-3337U @ 1,80Ghz, grafickou kartou Intel HD Graphics 4000 a operační paměti o velikosti 4 GB.

Naměřené průměrné hodnoty metod PSNR, SSIM a M-SVD u všech čtyř videí jsou zobrazeny na obr. 4.10 až obr. 4.21. Pokud bychom chtěli hodnoty PSNR přirovnat ke kategoriím ACR, více než 40 dB by přibližně odpovídalo kategorii 5, tedy Excellent. Hodnotu 35 dB lze obecně považovat za kvalitativní kategorii Good, záleží však zejména na použité scéně. Další kategorie pak budou odpovídat kroku -5 dB. U ostatních dvou metod nejsou tyto přibližné hodnoty lineárně rozmístěny, lze však říci, že u metody SSIM hodnoty větší než 0,95 znamenají vynikající kvalitu obrazu. U metody M-SVD jsou to zejména hodnoty menší než 3.

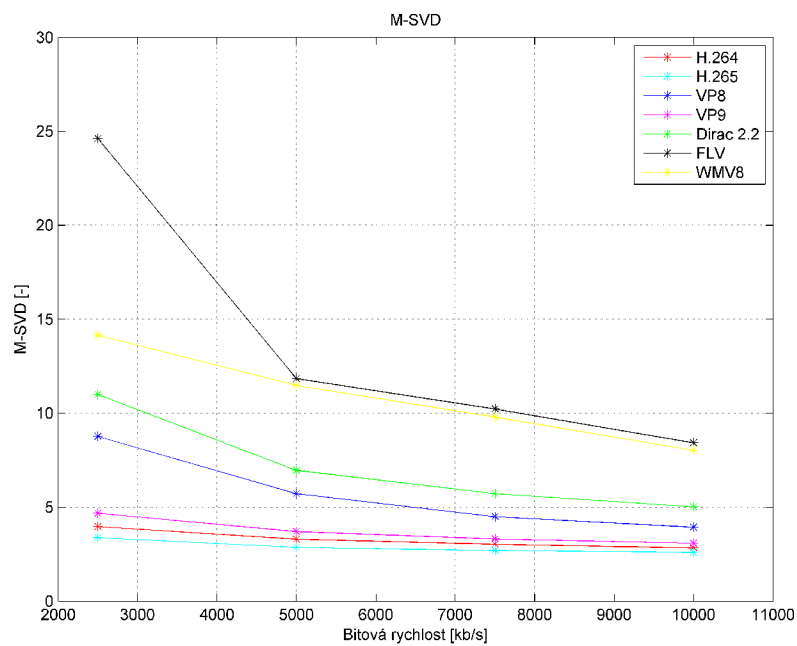
Ve všech provedených měřeních dosahoval jednoznačně nejlepších výsledků kodek H.265. U žádného videa ani bitové rychlosti neměl žádný kodek lepší průměrnou hodnotu PSNR, SSIM či M-SVD. U videa Mobcal a Life byl vždy v popředí s velkým odstupem také kodek H.264. Velice dobrých výsledků dosahoval kodek VP9, který u videa Blue sky vykazoval dokonce lepší hodnoty než kodek H.264 a srovnatelné hodnoty u videa Old town. Jeho velkou nevýhodou je však neúnosně dlouhá doba kódování ve srovnání se všemi ostatními kodeky, což může být způsobeno konkrétní, ne příliš výkonnou implementací v knihovně ffmpeg. Kodek VP8 dosahoval vždy průměrných výsledků s výjimkou videa Blue sky, při kterém byly jeho naměřené hodnoty, zejména PSNR (obr. 4.13) a M-SVD (obr. 4.15), srovnatelné s kodekem H.264. Kodek Dirac 2.2.3 značně převyšuje zbylé dva kodeky FLV a WMV8, avšak kromě hodnot M-SVD (obr. 4.18) a MSSIM (obr. 4.17) u videa Life, byl vždy horší než předchozí testované kodeky. Za jeho výhodu lze považovat velice krátkou dobu kódování i při rozlišení Full HD a snímkové rychlosti 50 fps. Kodeky WMV8 a FLV jsou v porovnání s ostatními kodeky již mírně zastaralé, což se projevilo také v naměřených hodnotách. Zejména u bitové rychlosti 2500 kb/s lze pozorovat prudký pokles obrazové kvality, což je způsobeno méně pokročilými technikami kódování, které nejsou schopné již více snížit bitovou rychlost, a proto dojde k neúměrnému zvýšení kvantizačního parametru a tím ke značné ztrátě informace. Nicméně kodek WMV8 vychází z měření lépe s výjimkou videa Blue sky, kdy byl ve všech třech testech nejhorší.



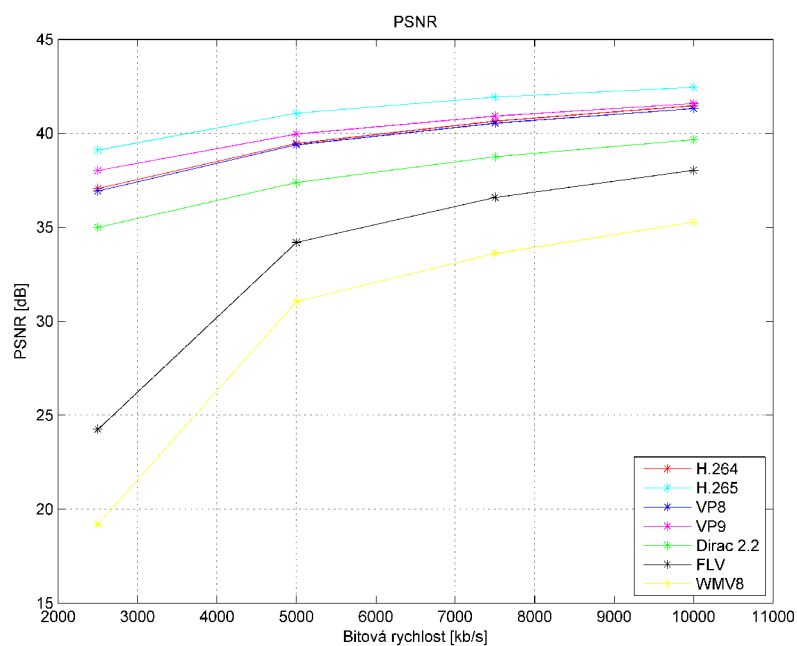
Obr. 4.10: Mobcal – průměrné hodnoty PSNR v závislosti na bitové rychlosti.



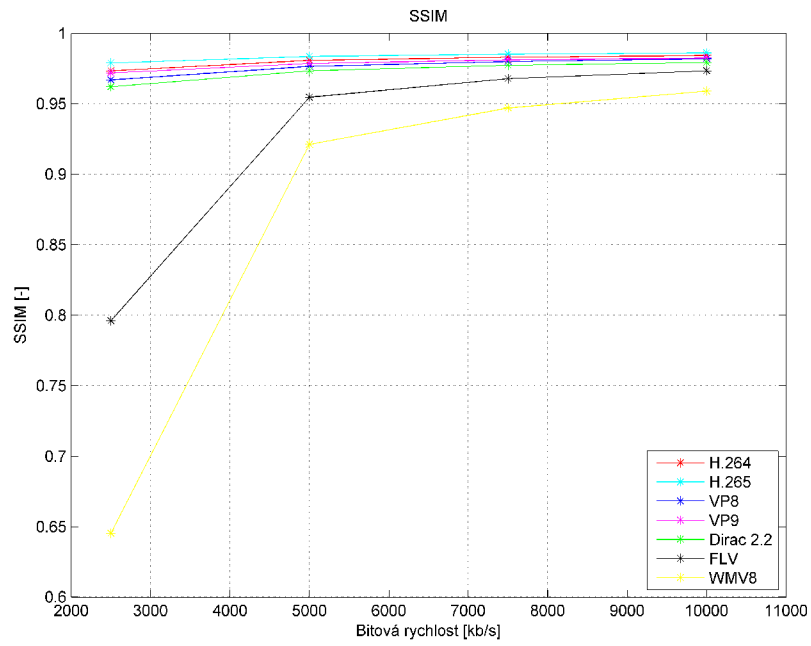
Obr. 4.11: Mobcal – hodnoty MSSIM v závislosti na bitové rychlosti.



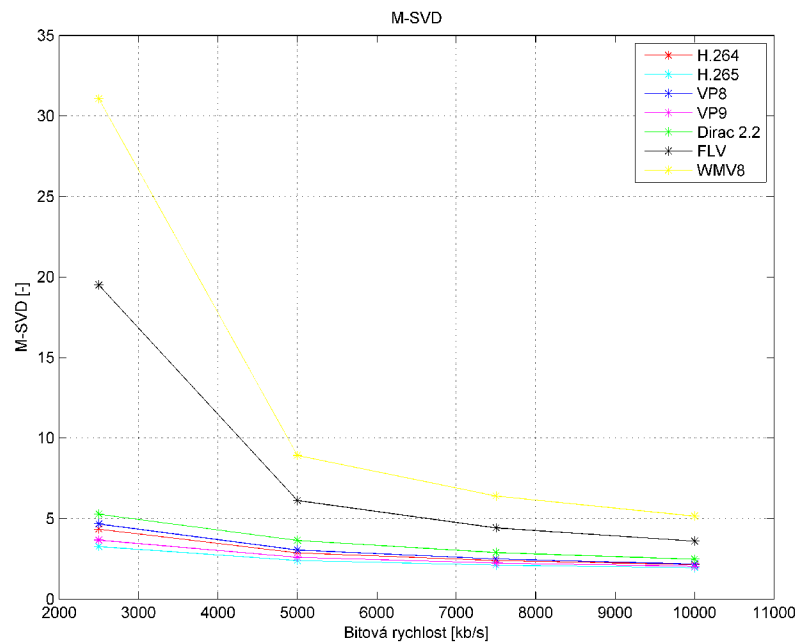
Obr. 4.12: Mobcal – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.



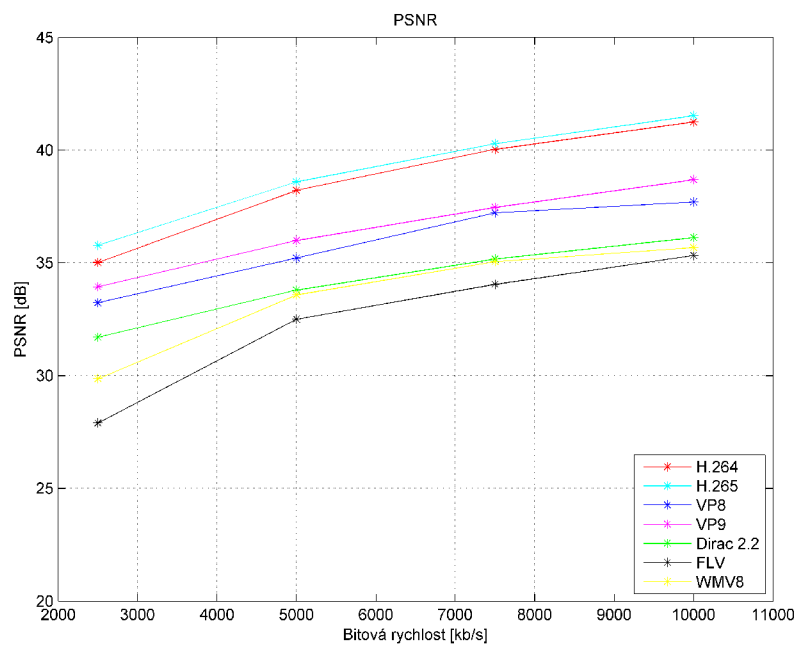
Obr. 4.13: Blue sky – průměrné hodnoty PSNR v závislosti na bitové rychlosti.



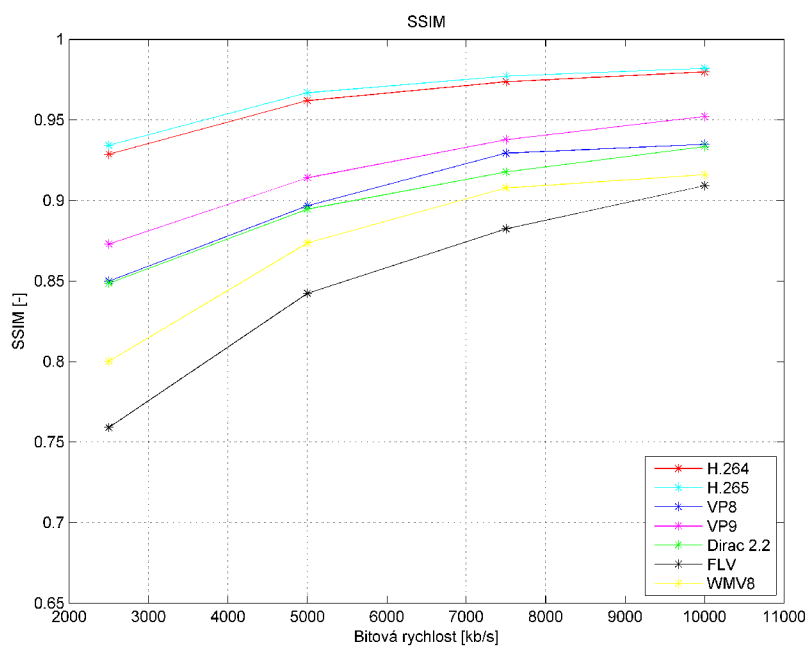
Obr. 4.14: Blue sky – hodnoty MSSIM v závislosti na bitové rychlosti.



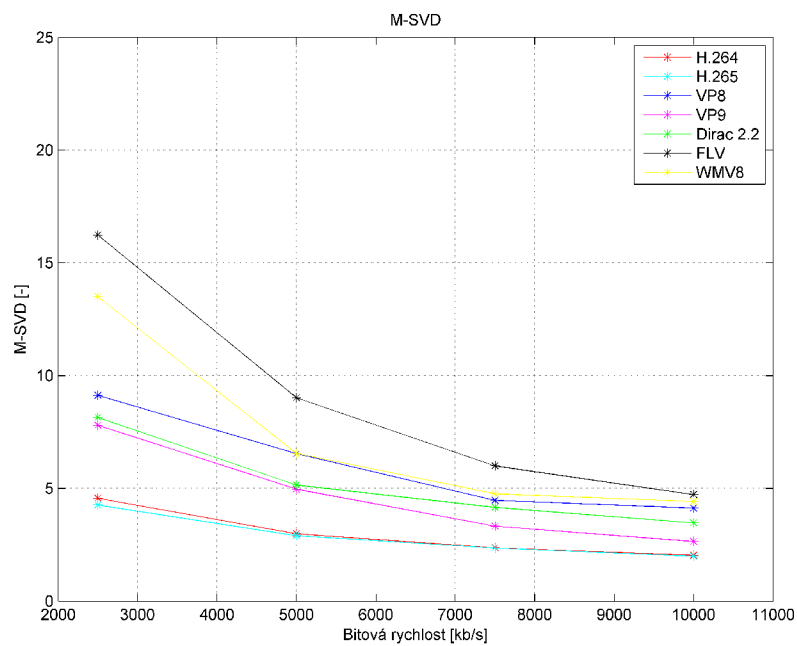
Obr. 4.15: Blue sky – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.



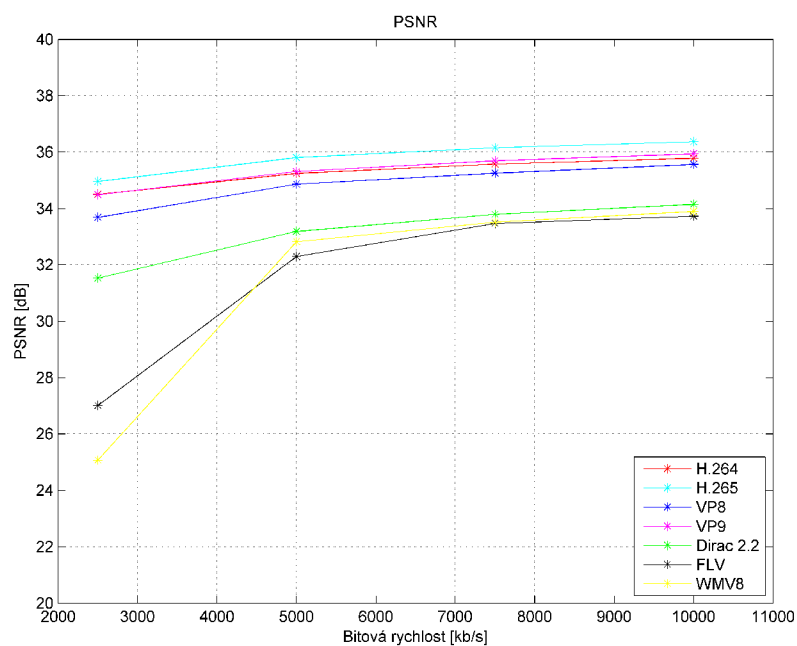
Obr. 4.16: Life – průměrné hodnoty PSNR v závislosti na bitové rychlosti.



Obr. 4.17: Life – hodnoty MSSIM v závislosti na bitové rychlosti.

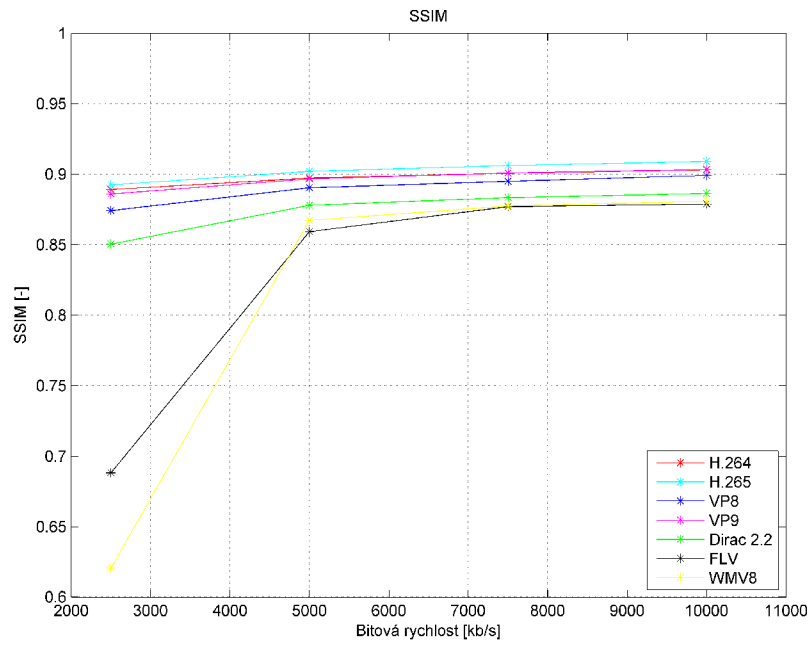


Obr. 4.18: Life – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.

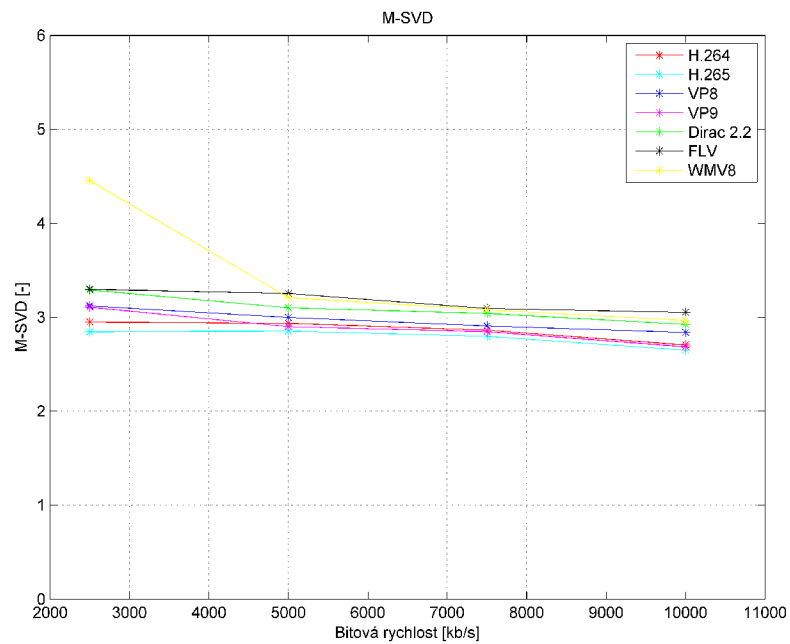


Obr. 4.19: Old town – průměrné hodnoty PSNR v závislosti na bitové rychlosti.





Obr. 4.20: Old town – hodnoty MSSIM v závislosti na bitové rychlosti.



Obr. 4.21: Old town – průměrné hodnoty M-SVD v závislosti na bitové rychlosti.

## 5 ZÁVĚR

V této bakalářské práci byly podrobně popsány objektivní metody hodnocení kvality videa, zejména pak metody PSNR, SSIM a M-SVD a vysvětleny principy měření kvality pomocí některých standardizovaných subjektivních metod. V další části byly popsány vlastnosti nejrozšířenějších kodeků H.264 a VP8, jejich nejnovějších nástupců H.265 a VP9 a dalších rozšířených alternativ, jako například Dirac či WMV.

V praktické části byla popsána aplikace VideoCodecs, která byla naprogramována v jazyce C++ jako součást této práce. Pomocí výše uvedené aplikace lze video zakódotovat do všech zmíněných formátů a také jej zpětně dekodovat. Aplikace dále umožňuje testování obrazové kvality videa pomocí výše uvedených metod PSNR, SSIM a M-SVD. Součástí aplikace je také simulace chybového kanálu, vhodná pro testování chybové odolnosti jednotlivých video kodeků.

Pomocí aplikace VideoCodecs, která uživatelsky zpřístupňuje funkce knihovny ffmpeg, byly na videích ve vysokém rozlišení otestovány všechny zmíněné kodeky, při různých bitových rychlostech výstupního videa. Ze srovnání jednotlivých kodeků vychází nejlépe nejnovější kodek H.265. Kodeky H.264 a VP9 mají často srovnatelnou kvalitu, avšak kodek VP9 disponuje velice pomalým kóděrem, což může být způsobeno konkrétní implementací v knihovně ffmpeg. Kodek VP8 dosáhl ve všech měřeních průměrných výsledků. Pokud hledáme flexibilní a zároveň kompaktní kodek s rychlým kóděrem, lze doporučit Dirac 2.2.3, který přes svoji méně rozsáhlou základnu nástrojů dosahuje velmi přijatelných výsledků. Ostatní dva kodeky FLV a WMV8 nedosáhly v tomto srovnání příliš dobrých výsledků a nelze je v současné době dále doporučit.

## LITERATURA

- [1] AKRAMULLAH, Shahriar. *Digital Video Concepts, Methods, and Metrics: Quality, Compression, Performance, and Power Trade-off Analysis*. New York: Apress Media, 2014. ISBN-13: 978-1-4302-6713-3.
- [2] BANKOSKI, James; WILKINS, Paul; XU, Yaowu. *Vp8 data format and decoding guide*. 2011. Dostupné z URL: <<http://www.rfc-editor.org/rfc/pdf/rfc6386.txt.pdf>>.
- [3] ALI, Irfan. *Bit-Error-Rate (BER) Simulation Using MATLAB*. *International Journal of Engineering Research and Applications (IJERA)*, 2013, 3: 706-711. Dostupné z URL: <[http://www.ijera.com/papers/Vol3\\_issue1/DB31706711.pdf](http://www.ijera.com/papers/Vol3_issue1/DB31706711.pdf)>.
- [4] BANKOSKI, Jim; WILKINS, Paul; XU, Yaowu. *Technical overview of VP8, an open source video codec for the web*. In: ICME. 2011. p. 1-6. Dostupné z URL: <<http://www.australianscience.com.au/research/google/37073.pdf>>.
- [5] EGIAZARIAN, Karen, et al. *New full-reference quality metrics based on HVS*. CD-ROM proceedings of the second international workshop on video processing and quality metrics, Scottsdale, USA. 2006. Dostupné z URL: <<http://www.comlab.uniroma3.it/Marco/Articoli/Battisti/A%20NEW%20FULL-REFERENCE%20QUALITY%20METRICS%20BASED%20ON%20HVS.pdf>>.
- [6] GRANGE, Adrian; ALVESTRAND, Harald. *A VP9 bitstream overview*. 2013. Dostupné z URL: <<http://tools.ietf.org/pdf/draft-grange-vp9-bitstream-00.pdf>>.
- [7] HEKSTRA, Andries P., et al. *PVQM – A perceptual video quality measure*. *Signal processing: Image communication*, 2002, 17.10: 781-798. Dostupné z URL: <<http://www.sciencedirect.com/science/article/pii/S0923596502000565#>>.
- [8] CHANDLER, Damon M.; HEMAMI, Sheila S. *VSNR: A wavelet-based visual signal-to-noise ratio for natural images*. *Image Processing, IEEE Transactions on*, 2007, 16.9: 2284-2298.
- [9] CHI, Chi Ching, et al. *Improving the parallelization efficiency of HEVC decoding*. *Image Processing (ICIP), 2012 19th IEEE International Conference on*. IEEE, 2012. p. 213-216. Dostupné z URL: <<http://iphone.hhi.de/schierl/assets/Parallelization-Efficiency-HEVC.pdf>>.

- [10] CRF Guide [online]. [cit.2015-5-16], Dostupné z URL: <<http://slhck.info/articles/crf>>.
- [11] ITU-R RECOMMENDATION, BT.500-13. Methodology for the subjective assessment of the quality of television pictures. 2012. Dostupné z URL: <<http://www.itu.int/rec/R-REC-BT.500-13-201201-I/en>>.
- [12] ITU-T RECOMMENDATION, P.910. Subjective video quality assessment methods for multimedia applications. 2008. Dostupné z URL: <<http://www.itu.int/rec/T-REC-P.910-200804-I/en>>.
- [13] KALVA, Hari; LEE, J.-B. *The VC-1 video coding standard. MultiMedia*, IEEE, 2007, 14.4: 88-91. Dostupné z URL: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4354161>>.
- [14] MUKHERJEE, Debargha, et al. *The latest open-source video codec VP9- An overview and preliminary results*. PCS. 2013. p. 390-393. Dostupné z URL: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6737765>>.
- [15] PETERSON, Heidi A.; AHUMADA JR, Albert J.; WATSON, Andrew B. *Improved detection model for DCT coefficient quantization*. IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology. International Society for Optics and Photonics, 1993. p. 191-201. Dostupné z URL: <<http://128.102.119.102/publications/spie93hap/DCTCoefficientQuantization.pdf>>.
- [16] POURAZAD, Mahsa T., et al. *HEVC: the new gold standard for video compression: how does HEVC compare with H. 264/AVC?*. Consumer Electronics Magazine, IEEE, 2012, 1.3: 36-46. Dostupné z URL: <[http://dml.ece.ubc.ca/doc/HEVC\\_2012.pdf](http://dml.ece.ubc.ca/doc/HEVC_2012.pdf)>.
- [17] RICHARDSON, Iain E. *The H.264 Advanced Video Compression Standard*. London: Wiley-Blackwell, 2010. ISBN 978-0470516928.
- [18] SHEIKH, Hamid R.; BOVIK, Alan C. *Image information and visual quality*. Image Processing, IEEE Transactions on, 2006, 15.2: 430-444. Dostupné z URL: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1576816&tag=1>>.

- [19] SHEIKH, Hamid R.; BOVIK, Alan C. *A visual information fidelity approach to video quality assessment*. The First International Workshop on Video Processing and Quality Metrics for Consumer Electronics. 2005. p. 23-25. Dostupné z URL: <<http://202.114.89.42/resource/pdf/415.pdf>>.
- [20] SMPTE, Proposed. 421M, *VC-1 Compressed Video Bitstream Format and Decoding Process*. 2006. Dostupné z URL: <<http://multimedia.cx/mirror/s421m.pdf>>
- [21] SPECIFICATION, BBC Dirac, Version 2.2.3., September 2008. Dostupné z URL: <<http://diracvideo.org/download/specification/dirac-spec-latest.pdf>>.
- [22] SULLIVAN, Gary J., et al. *Overview of the high efficiency video coding (HEVC) standard*. Circuits and Systems for Video Technology, IEEE Transactions on, 2012, 22.12: 1649-1668. Dostupné z URL: <[http://iphone.hhi.de/wiegand/assets/pdfs/2012\\_12\\_IEEE-HEVC-Overview.pdf](http://iphone.hhi.de/wiegand/assets/pdfs/2012_12_IEEE-HEVC-Overview.pdf)>.
- [23] *SWF File Format Specification*. Adobe, Version 19, 2012. Dostupné z URL: <<http://wwwimages.adobe.com/content/dam/Adobe/en/devnet/swf/pdf/swf-file-format-spec.pdf>>
- [24] ŠILHAVÝ, Pavel. *Datové komunikace*. Brno, 2012. ISBN 978-80-214-4455-3. Skripta. Vysoké učení technické v Brně. Dostupné z URL: <[https://www.vutbr.cz/www\\_base/priloha.php?dpid=68492](https://www.vutbr.cz/www_base/priloha.php?dpid=68492)>.
- [25] TAO, Peining; ESKICIOGLU, Ahmet M. *Video quality assesment using M-SVD*. Electronic Imaging 2007. International Society for Optics and Photonics, 2007. p. 649408-649408-10.
- [26] WANG, Zhou, et al. *Image quality assessment: from error visibility to structural similarity*. Image Processing, IEEE Transactions on, 2004, 13.4: 600-612. Dostupné z URL: <<https://ece.uwaterloo.ca/~z70wang/publications/ssim.pdf>>.
- [27] WANG, Zhou; BOVIK, Alan C. *Mean squared error: love it or leave it? A new look at signal fidelity measures*. Signal Processing Magazine, IEEE, 2009, 26.1: 98-117. Dostupné z URL: <<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4775883>>.

- [28] WANG, Zhou; LU, Ligang; BOVIK, Alan C. *Video quality assessment based on structural distortion measurement*. Signal processing: Image communication, 2004, 19.2: 121-132. Dostupné z URL: <[http://live.ece.utexas.edu/publications/2004/zwang\\_vssim\\_spim\\_2004.pdf](http://live.ece.utexas.edu/publications/2004/zwang_vssim_spim_2004.pdf)>.
- [29] WANG, Zhou; SHEIKH, Hamid R.; BOVIK, Alan C. *Objective video quality assessment. The handbook of video databases: design and applications*. 2003, 1041-1078. Dostupné z URL: <[http://atc.umh.es/gatcom/bin/oqam/Referencias/Wang\\_Sheikh\\_Bovik\\_BookChapter\\_2003.pdf](http://atc.umh.es/gatcom/bin/oqam/Referencias/Wang_Sheikh_Bovik_BookChapter_2003.pdf)>.
- [30] WANG, Zhou; SIMONCELLI, Eero P.; BOVIK, Alan C. *Multiscale structural similarity for image quality assessment*. Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on. Ieee, 2003. p. 1398-1402. Dostupné z URL: <<https://ece.uwaterloo.ca/~z70wang/publications/msssim.pdf>>.
- [31] WATSON, Andrew B.; HU, James; MCGOWAN, John F. *Digital video quality metric based on human vision*. Journal of Electronic imaging, 2001, 10.1: 20-29. Dostupné z URL: <[http://hsi.arc.nasa.gov/publications/jei2000\\_2wc.doc.pdf](http://hsi.arc.nasa.gov/publications/jei2000_2wc.doc.pdf)>.
- [32] *Xiph.org Video Test Media [derf's collection]*[online].[cit.2015-5-10], Dostupné z URL: <<https://media.xiph.org/video/derf/>>.
- [33] ZENG, Kai; WANG, Zhou. *3D-SSIM for video quality assessment*. Image Processing (ICIP), 2012 19th IEEE International Conference on. IEEE, 2012. p. 621-624. Dostupné z URL: <<https://ece.uwaterloo.ca/~z70wang/publications/ICIP12c.pdf>>.

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

ACR	Absolute Category Rating
ACR–HR	Absolute Category Rating – Hidden Reference
ADST	asymetrická diskretní sinová transformace – Asymmetric Discrete Sine Transform
BER	bitová chybovost – Bit Error Rate
CABAC	Context Adaptive Binary Arithmetic Coding
CAVLC	Context Adaptive Variable Length Coding
CBR	konstantní bitová rychlost – Constant Bit Rate
CCR	Comparison Category Rating
CRF	Constant Rate Factor
CSF	funkce kontrastní citlivosti – Contrast Sensitivity Function
CSV	hodnoty oddělené středníkem – Comma Separated Values
CTU	Code Tree Unit
DCS	Differential Category Rating
DSCQS	Double Stimulus Continuous Quality Scale
DMOS	Differential Mean Opinion Score
DCT	diskretní kosinová transformace – Discrete Cosine Transform
DPB	zásobník dekodovaných snímků – Decoded Picture Buffer
DST	diskretní sinová transformace – Discrete Sine Transform
DVQ	Digital Video Quality
DWT	diskretní vlnková transformace – Discrete Wavelet Transform
FLV	Flash Video
GOB	skupina bloků – Group Of Blocks
GOP	skupina snímků – Group of Pictures

GUI	grafické uživatelské rozhraní – Graphical User Interface
HVS	psychovizuální model lidského vidění – Human Visual System Model
MOS	Mean Opinion Score
MSE	střední kvadratická chyba – Mean Squared Error
MSSIM	průměrná hodnota SSIM – Mean Structural Similarity Index
MS-SSIM	Multi-Scale SSIM
PC	metoda porovnávání párů – Pair Comparison Method
PSNR	špičkový poměr signálu k šumu – Peak Signal-to-Noise Ratio
PVQM	Perceptual Video Quality Metric
SAC	Syntax-based Arithmetic Coding
SER	symbolová chybovost – Symbol Error Rate
SMPTE	Society of Motion Picture and Television Engineers
SSCQE	Single Stimulus Continuous Quality Evaluation
SSIM	Structural Similarity Index
SVD	rozklad na singulární hodnoty – Singular Value Decomposition
UHD	ultra vysoké rozlišení – Ultra High Definition
VBR	proměnná bitová rychlost – Variable Bit Rate
VIF	Visual Information Fidelity
VLC	kódování s proměnnou délkou kódu – Variable Length Coding
VSNR	Visual Signal-to-Noise Ratio
VSSIM	Video SSIM
WHT	Walsh-Hadamard Transform
WMV	Windows Media Video



# SEZNAM PŘÍLOH

A Obsah Přiloženého CD

53

## A OBSAH PŘILOŽENÉHO CD

Součástí přiloženého CD je elektronická verze této práce. Dále se zde nachází archiv *VideoCodecs.zip*, kde ve složce *src* můžeme nalézt všechny zdrojové kódy aplikace. Ve složce *bin* poté zkompileovanou aplikaci *VideoCodecs.exe* včetně všech nezbytných dynamických knihoven nutných pro běh této aplikace. Aplikace byla vytvořena použitím kompilátoru MSVC2013 32-bit a lze ji spustit na PC s operačním systémem Microsoft Windows 7 a novější. Použitý aplikační framework Qt je však multiplatformní, díky tomu je možné zdrojové kódy zkompileovat i pro jiné operační systémy.

Obsah:

- *Moderní video kodeky.pdf* – elektronická verze práce
- *VideoCodecs.zip*
  - *bin* – obsahuje binární soubory, pro spuštění *VideoCodecs.exe* nejprve rozbalte
  - *src* – obsahuje zdrojové kódy aplikace