

Metody komprese videa

Video compression methods

Milan Reichert



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2010/2011

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Milan REICHERT**
Osobní číslo: **A07201**
Studijní program: **B 3902 Inženýrská informatika**
Studijní obor: **Bezpečnostní technologie, systémy a management**

Téma práce: **Metody komprese videa**

Zásady pro vypracování:

1. Vypracujte literární rešerši na téma komprese videa. Vytvořte stručný přehled nejpoužívanějších kompresí a softwaru pro kompresi videa.
2. Shrňte teoretické poznatky o tvorbě WWW stránek (HTML, CSS, ...).
3. Otestujte vybrané metody komprese videa na vzorcích videa a porovnejte tyto metody komprese z hlediska stupně komprese, časové náročnosti a kvality výsledného videa.
4. Vytvořte WWW stránky, které budou obsahovat přehled nejpoužívanějších metod komprese videa a jejich porovnání.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. CASTRO, Elizabeth. HTML, XHTML a CSS. Národní průvodce tvorbou WWW stránek. 1. vydání. Brno: Computer Press, 2007. 440 s. ISBN 978-80-251-1531-2
2. ČÁPEK, Jan, FABIAN, Peter. Komprimace dat: Principy a praxe. 1. vydání. Brno: Computer Press, 2000. 190 s. ISBN 8072262319
3. FURHT, Borko. Encyclopedia of Multimedia. Is.I.J : Springer US, 2006. 983 s. ISBN 978-0-387-30038-2.
4. LEVICKÝ, Dušan. Multimediálne telekomunikácie. 1. vydání. Košice: Elfa, s.r.o. 2002. 241 s. ISBN 80-89066-58-5
5. SAYOOD, Khalid. Introduction to Data Compression. 3rd edition. Is.I.J : Elsevier Inc., 2006. 704 s. ISBN 978-0-12-620862-7.

Vedoucí bakalářské práce:

Ing. Michal Heczko

Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce:

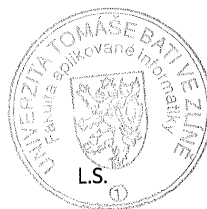
25. února 2011

Termín odevzdání bakalářské práce:

23. května 2011

Ve Zlíně dne 25. února 2011

prof. Ing. Vladimír Vašek, CSc.
děkan



doc. Mgr. Milan Adámek, Ph.D.
ředitel ústavu

ABSTRAKT

Tato bakalářská práce se zaměřuje na aktuálně používané formáty komprese videa. V první části jsou popsány základní principy těchto kompresních formátů. Následuje jejich porovnání a vyhodnocení. V závěru jsou veškeré poznatky shrnuty ve formě WWW stránky.

Klíčová slova: Komprese, Srovnání, Video, HTML, WWW.

ABSTRACT

This bachelor thesis is focused on currently used video compression formats. There is a basic description of the principles of compression formats in the first part. The following is the comparison and evaluation. In conclusion of this thesis, there is summary of the knowledge in the form of web pages

Keywords: Compression, Comparison, Video, HTML, WWW.

Chtěl bych poděkovat svému vedoucímu Ing. Michalu Heczkovi za veškerou pomoc a rady, které mi poskytl k úspěšnému dokončení této práce.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně 20. května 2011

.....
podpis diplomanta

OBSAH

ÚVOD.....	8
I TEORETICKÁ ČÁST.....	9
1 KOMPRESSE VIDEO	10
1.1 ZTRÁTOVÁ KOMPRESSE.....	11
1.1.1 MPEG.....	11
1.1.1.1 MPEG 1	12
1.1.1.2 MPEG 2	13
1.1.1.3 MPEG 4	15
1.1.2 H.261	17
1.1.3 H.263	18
1.1.4 H.264	19
1.2 BEZZTRÁTOVÁ KOMPRESSE.....	19
1.2.1 HuffYUV	19
1.2.1.1 Huffmanovo kódování	19
1.2.1.2 YUV	20
1.2.2 LCL	21
1.2.2.1 AVImszh.....	21
1.2.2.2 AVIzlib	21
1.2.3 Lagarith	22
2 TVORBA WWW STRÁNEK.....	23
2.1 HTML.....	23
2.1.1 Struktura HTML stránky.....	23
2.2 XHTML.....	25
2.3 CSS.....	25
2.3.1 Deklarace stylu.....	25
II PRAKTICKÁ ČÁST	27
3 KOMPRESSE VIDEO	28
3.1 SYSTÉM.....	28
3.2 POUŽITÉ VIDEO.....	28
3.3 POUŽITÉ PROGRAMY	29
3.3.1 VirtualDub 1.9.11.....	29
3.3.2 Any video convertor.....	30
3.4 VYHODNOCENÍ KOMPRESSE	31
4 TVORBA WWW STRÁNEK.....	34
ZÁVĚR	35
ZÁVĚR V ANGLIČTINĚ.....	36
SEZNAM POUŽITÉ LITERATURY.....	37
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	39
SEZNAM TABULEK A GRAFŮ.....	40
SEZNAM OBRÁZKŮ	41

ÚVOD

Úkolem této bakalářské práce bylo vytvoření podkladů pro seznámení se metodami komprese videa, a to s používanými i méně používanými, známými i neznámými, jakožto i jejich porovnání a zhodnocení, aby si každý mohl vybrat tu, která nejlépe vyhovuje jeho požadavkům. Zároveň v ní chci každému přiblížit významy a principy fungování jednotlivých kompresních metod a jejich použití. V teoretické části je sepsán seznam a základní informace o kompresních standardech a v části pro praktickou část jsou vybrané metody aplikovány na vzorku videa a následně budou porovnány mezi sebou, co se týče komprese, rychlosti komprese a její kvality.

Dále bych zde chtěl zmínit základní metody tvorby www stránek, na jejichž základech postavím svoji konečnou verzi s hodnocením kompresních metod ve formě webové stránky. Tady zmíním základní metody vytváření webových stránek. Samotné provedení je možné si prohlédnout v přiložených souborech.

I. TEORETICKÁ ČÁST

1 KOMPRESSE VIDEO

Na začátek je třeba položit si pár otázek. První z nich zní: „Proč komprimujeme?“ A v zápětí na to se nám nabídne další otázka: „Jak budeme komprimovat?“

Na první otázku jsou nejpravděpodobnější následující možnosti, a to: potřebujeme uložit velký objem dat našeho videa, buď na pevný disk nebo jiné přenosové médium, anebo toto video či více videí potřebujeme někomu poslat přes LAN nebo WAN k dalším krokům práce s videem.

A tímto se dostáváme k mé druhé otázce, a to: Jak? Zde je důležité to, jak moc kvalitní je třeba mít video pro další práci a zpracovávání. A tudíž i to, jestli nám k zpracování bude stačit komprimovaný formát, anebo po přenosu budeme potřebovat video v jeho původním stavu. To je taková naše první křižovatka, kde se rozhodneme, jakou cestou se dát a tím pádem, kterou z kompresních metod zvolit, jestli ztrátovou nebo bezztrátovou.

Pokud nám jde především o archivaci, je výhodnější použití kompresních metod ztrátových, jelikož dosahují daleko lepších kompresních poměrů než bezztrátové. Na druhou stranu bezztrátové metody bude nejideálnější použít pro kompresi určenou pro urychlení přenosové cesty k dalšímu zpracování a následné archivaci.

Pro představu zde uvedu tabulku, která uvádí přenosové rychlosti pro jednotlivé typy videa. Zdroj tabulky je publikace Multimediálne telekomunikácie:

Tabulka 1 - Přenosové rychlosti různých typů videa

Typ videa	Přenosová rychlost [kb/s]	
	Bez komprese	S kompresí
Video v reálném čase (128x120 op)(9bitů/op)(15snímků/s)	2 074	64
Video v reálném čase (128x240 op)(9bitů/op)(15snímků/s)	4 147	384
Video v reálném čase (128x240 op)(9bitů/op)(30snímků/s)	8 294	2 000
Video server s nízkou snímkovou frekvencí (352x240 op)(9bitů/op)(10snímků/s)	7 603	384
Video server s kvalitou VCR (352x240 op)(24bitů/op)(30snímků/s)	60 825	1 100
Video server studiové kvality (640x480 op)(24bitů/op)(30snímků/s)	221 184	4 000
Televize s kvalitou HDTV (1125řádků)(24bitů/op)(30snímků/s)	800 000	60 000 - 127 000

1.1 Ztrátová komprese

Jak už je zřejmé z názvu tyto kompresní metody způsobí určité ztráty některých informací ve videu, které už nebude možné získat z komprimovaného formátu zpět. Ovšem dosahujeme zde daleko lepších možností komprese než u metod bezztrátových.

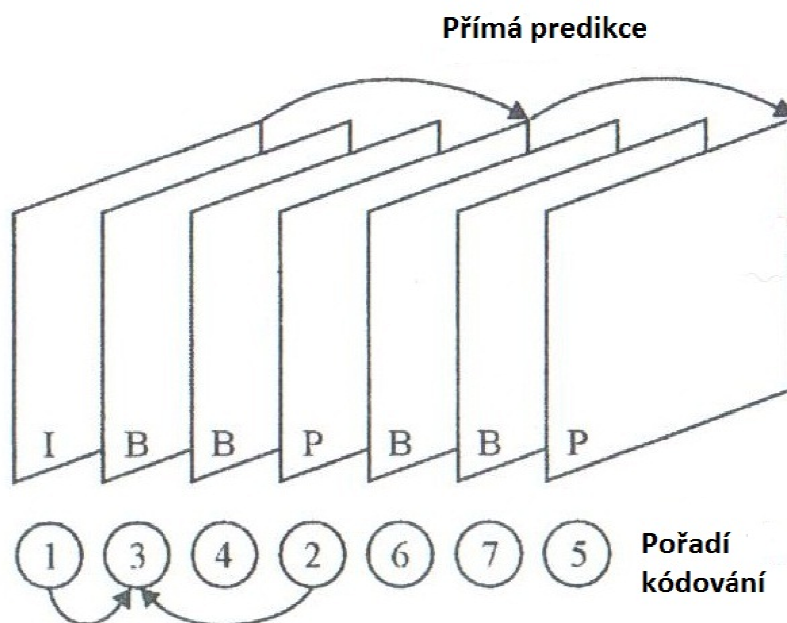
Největší výhodou ztrátové komprese je možnost úplného odstranění jistých částí obrazů, z nichž se video skládá, jako je například nepohyblivé pozadí. Dají se také snížit nároky na barevnost, jas a jiné složky obrazu, jež nám pomáhají snížit objem dat. Zmínil bych i nevýhodu – při vyšší kompresi je třeba počítat s výraznou ztrátou kvality.

1.1.1 MPEG

Toto je jedna z nejznámějších a nejpoužívanějších skupina kodeků. Písmena MPEG zkracují název expertní skupiny Motion Pictures Experts Group, která počátkem 90. let 20. století pracovala na standardizaci komprimačních formátů. V knize *Multimediálne telekomunikácie* je uvedená i následná struktura videosekvencí ve standardu MPEG: „*Na zabezpečení libovolného přístupu a rychlého přehrávání vpřed a vzad, jako i dalších režimů činnosti jsou definované tři druhy snímků:*

- *Snímky I s kódováním uvnitř snímku,*
- *Snímky P s primárním predikčním mezisnímkovým kódováním,*
- *Snímky B s obousměrnou predikcí/interpolací“*

Jednotlivé snímky jsou více popsány ve uvedené literatuře, ale pro přiblížení vám zde ještě zmíním něco o posloupnosti jednotlivých snímků: „*Organizaci posloupnosti snímků ve videosekvenci může uživatel libovolně měnit podle požadavků aplikace. Jestliže se použije posloupnost jen z I snímků (III...), tato posloupnost umožňuje vysoký stupeň libovolného přístupu editace a rychlého přetáčení vpřed, resp. Vzad, ale při nízké kompresi údajů což znamená velké nároky na přenosovou rychlost. Posloupnost složená z I a P snímků (např. IPPPPIPPP...) umožňuje určitý, resp. Menší stupeň libovolného přístupu a rychlost přetáčení při střední kompresi údajů a znamená nízké nároky na přenosovou rychlost. Tento typ posloupnosti využívá standard H.261 (více o něm níže). Posloupnost složená z I, P a B snímků umožňuje dosáhnout vysokou kompresi údajů a přiměřený stupeň libovolného přístupu a rychlého přetáčení, ale dost výrazně se zvyšuje zpoždění při kódování.*“ Příklad posloupnosti (viz. Obrázek 1).



Obrázek 1 - Příklad videosekvence

1.1.1.1 MPEG 1

Základní charakteristiky uvedené v publikaci Multimediálne telekomunikácie nám o tomto standardu říkají toto: „MPEG-1 video je standard pro zapamatování a vyhledávání pohyblivých obrazů na digitálních médiích v dobré kvalitě a při rychlosti okolo 1,4Mb/s a je určený pro aplikace s médii typu CD ROM.

Základní požadavky na tento standard byli formulované takto:

- Kódování videoinformace v dobré kvalitě od 1 do 1,5 Mb/s a audioinformace v dobré kvalitě od 128 do 256 kb/s
- Zastavení obrazu
- Libovolný přístup k snímkům v limitovaném čase
- Schopnost přehrávání vpřed a vzad rychlostí několikanásobně vyšší než je normální rychlost
- Synchronizace audio a videoinformace v režimu „playback“

Standard má být implementovaný v existujících dekodérech pracujících v reálném čase při vyhovujících cenových relacích.

Standard MPEG-1 video byl vyvinut s ohledem na existující standard JPEG a aktivity rozpracované při přípravě standardu H.261, přičemž jeho primární použití je v interaktivních multimediálních aplikacích na bázi paměťových médií CD-ROM. Parametry vstupních kódovaných obrazů je možné měnit flexibilně uživatelem, zejména prostorovou rozlišovací schopnost a snímkovou frekvenci, přičemž typickou obrazovou formou v MPEG-1 video je CIF.

Flexibilitu vstupních obrazů můžeme vyjádřit takto:

- Počet obrazových prvků v řádku menší nebo rovný 720
- Počet řádků obrazu menší nebo rovný 576
- Snímková frekvence menší nebo rovná 30
- Ve snímcích za sekundu
- Bitová rychlost menší nebo rovna 1,86 Mb/s

Základním principem v tomto standardu je hybridní mezisnímkové kódování, které využívá mezisnímkovou predikci s kompenzací pohybu kombinovanou s transformačním kódováním na bázi DCT. Základní standard MPEG-1 je založen na struktuře makrobloků. Vstupní barevný obraz se rozdělí na nepřekrývající se makrobloky, přičemž každý makroblok si nese informaci ve formátu YUV a to 4:1:1

Standard MPEG definuje tyto tři typy makrobloků:

- **Vynechaný makroblok** – je makroblok s malým vektorem pohybu. O tomto makrobloku se nepředává žádná informace,
- **Mezisnímkový makroblok** – je makroblok s nenulovým vektorem pohybu. Přenáší se informace o tomto typu makrobloku, jeho adresa, informace o vektoru pohybu a DCT koeficienty,
- **Makroblok uvnitř snímku** – je makroblok získaný bez mezisnímkové predikce. Přenáší se informace o typu makrobloku, jeho adresa a DCT koeficienty. “

1.1.1.2 MPEG 2

K tomuto standardu se uvádí v Multimediálně telekomunikácie toto: „Standard MPEG-2 je určen pro digitální televizi a také pro jiné aplikace zahrnující digitální kabelový přenos videa, satelitní televizi, přenos videa přes ATM síť, pro televizi a kvalitou HDTV.

Základní vlastnosti standardu MPEG-2 video:

- *Rozšíření MPEG-1 pro video s prokládaným řádkováním,*
- *Optimalizace pro televizní rozlišovací schopnost,*
- *Kvalita obrazu porovnatelná s normami NTSC, PAL, SECAM při přenosové rychlosti 4-8 Mb/s,*
- *Kvalita HDTV při přenosové rychlosti 20 Mb/s,*
- *Kompatibilita s vlastnostmi MPEG-1 video.*

Nové vlastnosti při kódování videa v MPEG-2 se získali zejména možností efektivního kódování videa s prokládaným řádkováním a škálovatelností (scalability).

Škálovatelnost umožňuje měnit rozlišení v oblastech poměru SNR (SNR Scalability), v prostorové oblasti (Spatial Scalability) a v časové oblasti (Temporal Scalability).

Do standardu MPEG-2 jsou zavedeny 2 nové pojmy:

- *Úroveň (level)*
- *Profil (profile)*

Tyto pojmy jsou prostředky pro různé implementace a pro vlastnosti dekodérů. Každá úroveň definuje množinu parametrů určených pro aplikaci (formát obrazu, snímková frekvence, přenosová rychlost, atd....). Profil určuje algoritmy a funkce podporující daný profil.“

Tabulka 2 - Mezní parametry úrovní v MPEG-2

Úroveň (LEVEL)	Parametry
Vysoká (HIGH)	1920 op/řádek 1152 řádků/snímek 60 snímků/s 80Mb/s
Vysoká 1440 (HIGH 1440)	1440 op/řádek 1152 řádků/snímek 60 snímků/s 60Mb/s
Hlavní (MAIN)	720 op/řádek 576 řádků/snímek 30 snímků/s 15Mb/s
Nízká (LOW)	352 op/řádek 288 řádků/snímek 30 snímků/s 4Mb/s

Tabulka 3 - Algoritmy pro profily v MPEG-2

Profil (profile)	Algoritmy
Vysoký (high)	podporují všechny funkce poskytované profilem "Prostorové škálování" doplněné o režimy kódování s 3 vrstvím prostorovým a SNR škálováním model YUV 4:2:2 pro náročné aplikace
Prostorové škálování (Spatial Scalable)	podporují všechny funkce poskytované profilem "SNR škálování" doplněné o režim kódování s 2 vrstvím prostorovým škálováním model YUV 4:2:0
SNR škálování (SNR Scalable)	podporují všechny funkce poskytované profilem "Hlavní škálování" doplněné o režim kódování s 2 vrstvím škálováním model YUV 4:2:0
Hlavní (Main)	Algoritmus bez škálování podporující kódování videa s prokládaným řádkováním, libovolná přístup k snímkům, režim predikce s využitím B snímků model YUV 4:2:0
Jednoduchý (Simple)	podporují všechny funkce poskytované profilem "Hlavní škálování" s výjimkou režimu predikce s využitím B snímků model YUV 4:2:0

1.1.1.3 MPEG 4

K tomuto standardu je v Multimediálně telekomunikácie uvedené následující: „MPEG-4 je standard pro multimediální aplikace, přičemž původně byl určený na kódování videa pro velmi nízké přenosové rychlosti, které bylo modifikováno na kódování audiovizuálních objektů. Jako příklady aplikací s využitím standardu MPEG-4 můžu uvést:

- Video na internetu,
- Video v mobilních sítích,
- Video-mail a domácí kino,
- Hry na bázi virtuální reality,
- Multimediální datové objekty.

Uvedené aplikace si vyžádali doplnění existujících standardů MPEG a tyto vlastnosti:

- **Interaktivnost na bázi obsahu** - (kontent based interactivity) je schopnost interakce s důležitými objekty scény. V standardu MPEG-4 je definováno více typů interakcí a to interakce syntetický (generovaných programem) objektů s reálnými objekty jako taky interakce s hybridními, tj. synteticko-reálnými objekty, které umožňují nové audiovizuální aplikace. Tato schopnost umožňuje také prostorové a časové škálování objektů.
- **Univerzální přístup k audiovizuálním datům** - (universal accessibility) je možnost přístupu k audiovizuálním datům přes různé paměťové a přenosové média. Vzhledem k tomu že trend směřuje k aplikacím v mobilních sítích je v standardu vytvořena možnost i přes tyto sítě.
- **Účinná komprese** - (improved compression) umožňuje zvýšit efektivitu přenosu a snížit nároky na kapacity paměťových médií. Protože standard MPEG-4 má schopnost manipulace s objekty, toto umožňuje flexibilně přizpůsobovat činitel komprese k šířce přenosového kanálu a ke kapacitě paměťového média. Když MPEG-4 využívá všechny dlouho známé metody komprese, kódování v tomto standardu bylo optimalizováno zejména pro nízké přenosové rychlosti do 64kb/s. Důležitou vlastností je také to, že je to kódování se zvýšenou robustností určené pro použití v podmínkách a velkým rušením a při nízkých přenosových rychlostech. Přenosová rychlost v MPEG-4 je v rozmezí 5-64kb/s pro aplikace v mobilních a v PSTN sítích a do 4Mb/s pro přenos TV resp. Filmových obrazů.

Z hlediska kódování videoinformace se v standardu MPEG-4 vyčlenili tyto dva druhy videoinformací:

- Reálné video (natural video)
- Syntetické video (synthetic video)

Kódování reálné videoinformace resp. Reálného videa je založeno na kódování metodou predikce s kompenzací pohybu na bázi DCT a na koncepci kódování videoobjektů.

Kódování syntetického videa se zakládá na modelování objektů s využitím zejména mřížkou modelované tvary (wire-frame objects) a různých přístupů k modelovanému objektu.“

Součástí MPEG-4 je nyní rozvíjející formát Mp4 jež bývá v některých případech uváděn jako MPEG-4 part 14.

1.1.2 H.261

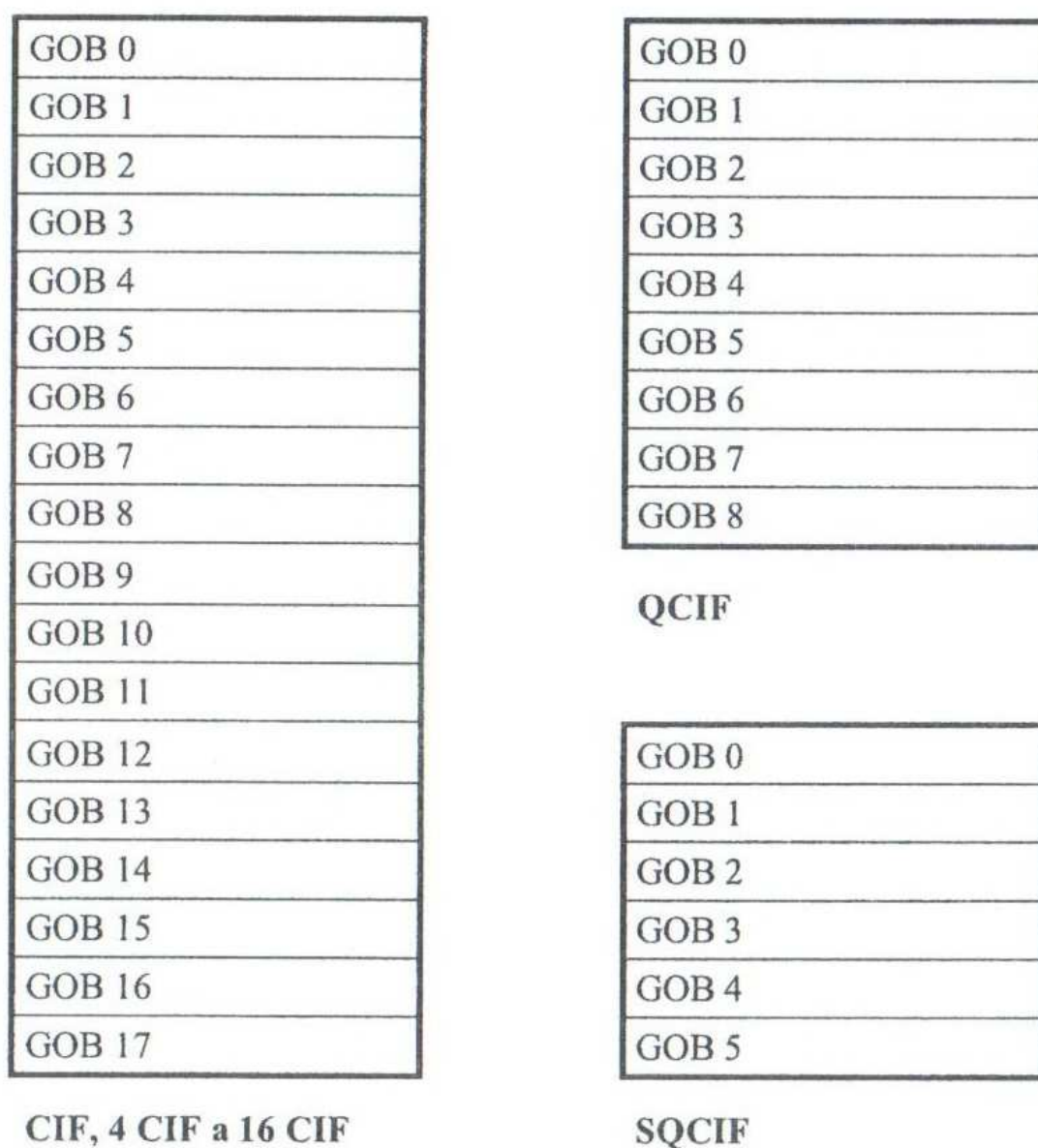
K tomuto video formátu a jeho struktuře bych rád uvedl následující informace z publikace Multimediálne telekomunikácie: „Standard H.261 byl navržen pro dva typy videoformátů a to CIF a QCIF. Uvedené formáty jsou potom dělené na menší části, které umožňují blokový přístup v kódování. Každý z uvedených formátů se dělí na skupiny bloků GOB (Groups of Blocks). Video formát se dělí na 12 skupin bloků GOB a formát QCIF na 3 skupiny bloků. Každý GOB se potom dělí na 33 makrobloků. Jednotlivé makrobloky zahrnují informaci ve formě $Y C_B C_R$. Kódovací algoritmus video standardu využívá hybridní blokové kódování, které je založené na mezisnímkové predikci a kompenzací pohybu, kombinované s transformačním kódováním na bázi DCT. Mezisnímková predikce s kompenzací pohybu odstraňuje časovou redundanci a transformační kódování redukuje prostorovou redundanci.“ Struktura videosekvence obsahuje pouze snímky I a P. přičemž snímek I je pouze první snímek. Snímky typu B se v standardu H.261 vůbec nepoužívají.

CIF		QCIF	
GOB 1	GOB 2	GOB 1	
GOB 3	GOB 4	GOB 3	
GOB 5	GOB 6	GOB 5	
GOB 7	GOB 8		
GOB 9	GOB 10		
GOB 11	GOB 12		

Obrázek 2 - Video formáty a jejich struktura H.261

1.1.3 H.263

Původním určením tohoto standardu bylo vytvořit kompresní formát pro videokonference. Standard H.263 byl vyvinut jako aktualizace H.261, díky zkušenostem a poznatkům získaných při vývoji algoritmů standardů MPEG a H.262, což vedlo k redukci přenosové rychlosti až o 50% oproti standardu H.261 při stejné kvalitě obrazu. Standard H.261 u něhož bylo ke kompresi využito makrobloků a nijak jinak tomu není ani u standardu H.263. Oproti H.261 byli také upraveny a přidány videoformáty (viz Obrázek 3). A také se začalo v struktuře video signálu používat B snímky.



Obrázek 3 - Video formáty a jejich struktura H.263

1.1.4 H.264

Jedná se o nejnovější standard ze skupiny H.26x. vychází z MPEG-4 part10. Více o tomto standardu viz Seznam literatury [15].

1.2 Bezztrátová komprese

Autor knihy *Multimediálne telekomunikácie* formuluje takto: „Bezztrátové kódování, resp. Entropické kódování je kódování s proměnlivou délkou kódového slova, proto se označuje jako VLC kódování (Variable length coding). VLC kódy jsou bezztrátové kódy, které zabezpečují rekonstrukci signálů bez degradace, jejich nevýhodou je však to, že poskytují kompresy pouze ve smyslu průměru. To znamená, že VLC kód může být pro některé části signálu neefektivní a VLC kodér produkuje průměrný tok bitů na svém výstupu. V případě přenosu konstantní přenosovou rychlostí to vyžaduje použití vyrovnávací paměti a řízení bitového toku kodéru. Podobně jako u kódování po blocích (rámcích), resp. Paketech je při použití VLC kódování obtížnější.“

Tyto kompresní metody nám dávají výhodu opětovného obnovení do původní kvality videa před kompresí. Ovšem komprimované video nedosahuje tak dobrého kompresního poměru jako video komprimované ztrátově.

1.2.1 HuffYUV

Název tohoto video kodeku je složen ze dvou názvů, jež popisují principy, na kterých kodek pracuje. Jsou to Huff, jež je označení pro Huffmanovo kódování a YUV, jež je označení barevného prostoru.

1.2.1.1 Huffmanovo kódování

Internetová encyklopedie Wikipedie k tomuto heslu uvádí: „Konvertuje znaky vstupního souboru do bitových řetězců různé délky. Znaky, které se ve vstupním souboru vyskytují nejčastěji, jsou konvertovány do bitových řetězců s nejkratší délkou (nejfrekventovanější znak tak může být konvertován do jediného bitu) znaky, které se vyskytují velmi zřídka, jsou konvertovány do delších řetězců (mohou být i delší než 8 bitů). Nejjednodušší metoda komprese touto metodou probíhá ve dvou fázích. První projde soubor a vytvoří statistiku četností každého znaku. Ve druhé fázi se využije této statistiky pro vytvoření binárního stromu a k následné kompresi vstupních dat. Dekomprese naopak pomocí rekonstruovaného binárního stromu dekóduje řetězce proměnlivé délky.“

1.2.1.2 YUV

V knize Moderní počítačová grafika tento prostor popisuje autor takto: “Základní prostor YUV se používá pro přenos televizních signálů v normě PAL. Do stejné rodiny patří další prostory – YIQ pro americkou televizní normu NTSC a prostor $Y C_B C_R$ pro normu SECAM. Jejich společným rysem je oddělení jasové (laminační) složky od barevných informací (chrominance). Toto dovoluje přímo využít stejný jasový signál Y jak pro barevné tak i pro černobílé televizory. Zbývající dva signály nesou informace o velikosti barevných složek obrazu.

Prostor YUV je někdy označován též $[Y, B-Y, R-Y]$, resp. Číselně $[Y, 0.493(B-Y), 0.877(R-Y)]$. Z prostoru RGB získáme hodnoty YUV jednoduchým maticovým násobením.”

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,141 & -0,289 & 0,437 \\ 0,615 & -0,515 & -0,100 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

YUV se navzájem od sebe ještě mohou lišit tím, kolik informací je v oblasti jasu a kolik je barevných informací. Tímto způsobem můžeme tento barevný prostor rozdělit na další části. Např.: YUV 2:1:1, který má 2 složky jasové a pak po jedné složce z barevných informací. Další možné jsou 2:1:1, který má 2 složky jasové a pak po jedné složce z barevných informací. Další možné jsou YUV 1:1:1, YUV 4:1:1, YUV 4:2:0.

Zde vám nyní pro porovnání uvedu obrázky jednotlivých YUV schémat jak jsou také k nalezení na stránce <http://www.geocities.co.jp/Playtown-Denei/2837/LRC.htm>:



Obrázek 4 - YUV 1:1:1



Obrázek 5 - YUV 4:2:2 (2:1:1)



Obrázek 6 - YUV 4:1:1



Obrázek 7 - YUV 4:2:0

1.2.2 LCL

Tento kodek byl dříve označován jako LRC. Je to freeware, vyvíjený Japoncem, jménem Kenji Oshima. Je určen pro operační systémy Windows a je schopný pracovat s barevným schématem RGB i YUV. Tento kodek nám dává výběr ze dvou možných způsobů komprimace a jsou to tyto dva: AVIzlib a AVImsh.

1.2.2.1 AVImsh

Tento kodek vychází z algoritmu LZ77, který v roce 1977 zveřejnili jeho autoři Lempel a Ziv do dnešní doby se dochoval jen jako základ, který byl mnohokrát modifikován. Kniha Základy digitální komunikace nám hlavní princip komprese popisuje takto: „*Algoritmus používá tzv. Posuvné okno, skrze které nahlíží na vstupní soubor. Tímto oknem přejede postupně přes celý původní soubor. Okno je rozděleno na dvě části: prohledávané pole a komprimované pole. Když algoritmus najde v komprimovaném poli posloupnost symbolů (řetězec), která se vyskytuje i v prohledávaném poli, do výstupního souboru nekopíruje tuto nalezenou posloupnost, ale zapíše pouze informaci o tom, kde v prohledávaném poli nalezená posloupnost začíná a jak je dlouhá. A na výstup také zkopíruje symbol bezprostředně následující po této (nekopírované) posloupnosti.*“

1.2.2.2 AVIzlib

Tento typ kódování, postaven z open source knihovny zlib. Zlib knihovna využívá ke kompresi metodu, zvanou DEFLATE, což je vlastně kombinace LZ77 a Huffmanova kódování. Blíže k této metodě bych uvedl jen následující. Soubor je opět rozdělen do bloků

o libovolné velikosti, kromě nekomprimovatelných bloků, jejichž velikost je omezena na 65,535 bytů. Huffmanovo kódování se používá na bloky nezávisle na předcházejících a následovných, tyto jsou následně zkomprimovány pomocí LZ77 jakožto řetězce znaků Huffmanových kódů.

1.2.3 Lagarith

Je to freeware video kodek s otevřeným kódem, jeho původní určení, podle autora, je zejména pro editaci a archivaci. Kódování vychází z Huffvuv, který používá jako šablonu a díky modifikacím jej v konečném důsledku překonává jak v kompresním poměru, tak i v čase. Takto to podává autor tohoto kodeku, jestliže je tomu skutečně tak, se přesvědčíme v praktické části. Hlavní rozdíl je v tom, že Lagarith používá na kompresi řetězce dat (byte stream) modifikované RLE (Run Length Encoding), oproti Huffvuv, používající Huffmanovo kódování. Tímto právě dosahuje lepšího zkomprimování dat. Použitím frakčních bitů na symbol umožňuje mít komprimovanou velikost velmi blízko k entropii dat, proto může Lagarith komprimovat jednotlivé obrazy. K tomuto je Lagarith opatřen podporou pro nulové obrazy (null frames) v případě, že je předchozí snímek totožný a se snímkem aktuálním, využije kódér předchozí snímek a ten aktuální vypustí.

2 TVORBA WWW STRÁNEK

K vytváření WWW stránek není třeba žádný speciální program, i když takové existují, jako příklad zde mohu uvést tyto - Microsoft Expression Web, Adobe Dreamweaver apod. Tyto využívají způsob editace zvaný WYSIWYG z anglického „What you see is what you got“, při kterém je verze na obrazovce vzhledově totožná s výsledným dokumentem. Je nutné ale uvést, že takovéto programy nám v mnoha případech mohou značně usnadnit práci. Pokud nemáme možnost specializovaného programu, postačí nám jakýkoliv textový editor např.: poznámkový blok. Aspoň tedy pro HTML.

2.1 HTML

HTML znamená Hypertext Markup Language - hypertextový (hypertext = odkaz) značkovací jazyk. Pomocí jazyka HTML a pomocí HTML značek (nebo také tagů) se tvoří webové stránky. Značek je mnoho a vždy se zapisují do špičatých závorek. Tyto značky rozdělují na dvě skupiny, jsou to značky párové a nepárové. Již z názvů značek vyplývá, v čem se od sebe liší. Párové značky mají vždy značku začáteční a koncovou značku. Nejlépe je to vidět na příkladu - `<head>` nám značí začátek hlavičky a `</head>` její konec. Ze zde uvedeného si můžeme vyvodit to, že značky nepárové nemají koncovou značku. Opět uvedu příklad: značka `<img>` je tagem pro obrázek, ale koncový tag typu `</img>` neexistuje.

2.1.1 Struktura HTML stránky

Vždy je nutné dodržovat stanovenou strukturu dokumentu. Standardní struktura dokumentu vypadá následovně:

```

<html>

  <head>

    hlavička stránky

    <title>Titulek stránky</title>

  </head>

  <body>

    obsah stránky

  </body>

</html>

```

Do souboru můžeme vložit také specifikaci DTD. Toto se píše úplně na začátek souboru ještě před <html>. Tato specifikace se udává v tomto stylu <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">. V následující tabulce uvádím příklady možných tagů.

Tabulka 4 - Příklady tagů

Tag	význam	Párový	skupina
b	tučné písmo	Ano	úprava textu
i	kurzíva	Ano	úprava textu
sub	dolní index	ano	úprava textu
font	písmo	ano	úprava textu
basefont	základní font	ne	úprava textu
p	odstavec	ano	bloky
br	řádkový zlom	ne	bloky
center	vycentrování	ano	bloky
ul	odrážkový seznam	ano	seznamy
menu	typ seznamu	ano	seznamy
img	obrázek	ne	obrázky
map	klikací mapa	ano	obrázky
script	skript, program	ano	skripty a styly
style	zápis CSS stylu	ano	skripty a styly
bgsound	zvuk na pozadí	ano	rozšíření
marquee	běžící text	ano	rozšíření
spacer	prázdné místo	ne	rozšíření

Další používané značky lze nalézt na internetu či v odborné literatuře.

2.2 XHTML

Tuto základní specifikaci uvádí autor na stránkách Tvorba webu: „*XHTML je značkovací jazyk. XHTML soubor má koncovku html nebo htm. Nyní se pro webové stránky používá právě XHTML (dříve HTML), který vychází z obecného standardu pro výměnu dat XML. XHTML popisuje význam obsahu (srovnání HTML a XHTML).*“ Struktura dokumentu je zde naprosto stejná jako u HTML, jelikož se z něho vycházelo při definování tohoto jazyka. Proto značky a jejich použití jsou totožné jako u HTML.

2.3 CSS

CSS (Cascading Style Sheets) kaskádové styly, které se používají k vytvoření stylu webové stránky (barva, písmo, velikost písma), jsou nadstavbou jazyka HTML. S CSS styly můžeme pomocí jednoho souboru ovlivňovat design celého webu. CSS styly se aplikují hlavně pomocí tříd a identifikátorů. Objekty v HTML dokumentu, jejichž třída nebo identifikátor odpovídají těm z definic v CSS, potom přebírají ty vlastnosti definované v CSS. CSS umožňují přiřadit každému prvku HTML určité vlastnosti (vzhled, způsob zobrazení, umístění ...), a to v závislosti na druhu zařízení, na němž je WWW dokument prezentován (obrazovka počítače, kapesní počítač a jiné). Díky CSS se na jeden element může navrstvit mnoho různých deklarací (proto styly kaskádové), někdy i protichůdných. Jak se rozhodne, která deklarace převládne? Zpravidla ta poslední. Pokud ovšem chceme, aby nějaká dřívější deklarace převládla, je třeba napsat do deklarace řetězec "!important". Taková deklarace potom nebude přehlušena žádnou pozdější. (Tuto vlastnost se doporučuje používat s rozvahou, protože způsobuje zmatek při pozdějším ladění designu.)

2.3.1 Deklarace stylu

Takto vypadá kód pro deklaraci stylu:

```
<style type="text/css">
.jméno_třídy { deklarace }
#jméno_identifikátoru { deklarace }
jméno_tagu { deklarace }
</style>
```

Další možnosti deklarací a využití stylů viz. Literatura [13],[14] či jiné tištěné nebo internetové zdroje.

II. PRAKTICKÁ ČÁST

3 KOMPRESSE VIDEOA

V této části se podíváme na jednotlivé programy, využívající výše popsané kompresní metody. Výsledky budou zaznamenány a uvedeny v tabulce na konci kapitoly. Obrazové záznamy pro zhodnocení kvality obrazu budou umístěny v přílohách a pod patřičnými názvy.

3.1 Systém

- Procesor: Intel CPU T2400 @ 1,83GHz
- RAM: 1 GB
- Grafická karta: ATI Mobility Radeon X1400
- Operační systém: Windows 7 Ultimate 32bit

Toto zde uvádím proto, aby bylo jasné, na jakém počítači bylo testování prováděno, jelikož výkonnost systému hraje velikou roli v tom, jak rychle zvládne program zpracovat požadavek na kompresi videa. Můj počítač není nejvýkonnější, a proto i následující programy nemusí pracovat tak jako na jiných počítačích. Na to je třeba brát zřetel u hodnocení výsledků.

3.2 Použité video

K testování jsem vybral video, které se mi podařilo stáhnout ze stránek <http://www.cdvl.org/index.php>. Což je projekt vytvářející knihovnu digitálních videí určená pro výzkumníky a vývojáře v oblasti zpracování videa a vizuální kvality. Jeho specifikace je následná:

- Velikost: 3,44 GB (3 697 982 828 bajtů)
- Rozlišení: 1280x720
- Rychlost dat: 737292 kbps
- Celková přenosová rychlost: 739596 kbps
- Frekvence snímků: 50 snímků/s
- Formát: avi

Video bude zařazeno mezi přílohy této práce.

3.3 Použité programy

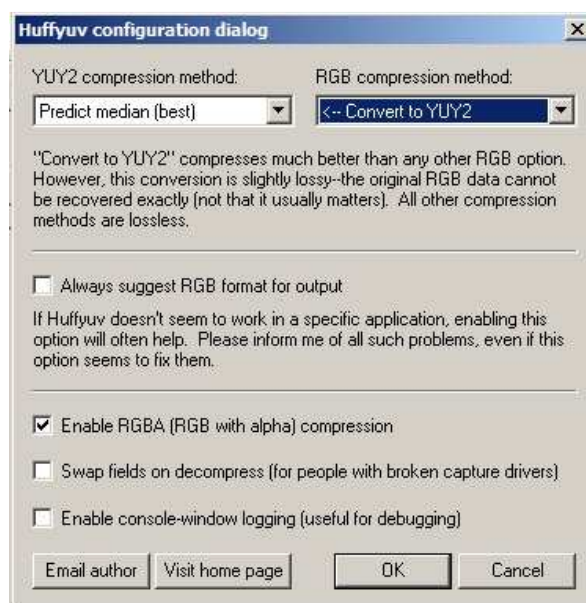
K testování kompresních algoritmů jsem vybral programy většinou volně dostupné na internetu, a pokud se jedná o placený program, využiji nyní pouze jeho zkušební dobu.

3.3.1 VirtualDub 1.9.11

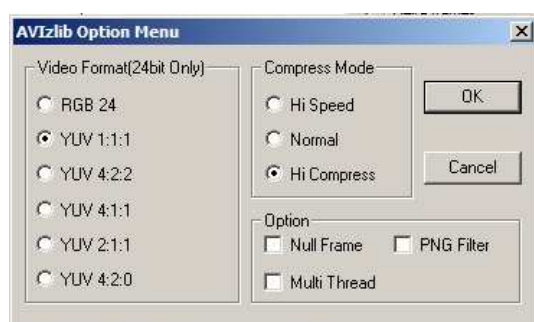
Tento program je volně stažitelný z internetové stránky <http://www.virtualdub.org> v sekci download. Tento program je určen pro práci s videem a umožňuje nám provádět kompresi vybranými metodami. Ty je ovšem potřeba do něho nahrát, tedy ne přímo do programu, ale do počítače je třeba umístit příslušné knihovny.

- Knihovnu pro HuffYUV 2.1.1 lze bezplatně stáhnout na stránkách:
<http://neuron2.net/www.math.berkeley.edu/benrg/huffyuv.html>, které obsahují i další potřebné informace co se týče instalace a nastavení.
- Knihovnu pro LCL 2.2.3 lze opět bezplatně stáhnout na internetu, na domácích stránkách projektu. Adresa je následovná:
http://www.geocities.jp/sandk_project/LRC.htm. popřípadě stránky v češtině přeložené přes translate.google.com zde -
<http://translate.google.com/translate?hl=en&sl=ja&tl=cs&u=http%3A%2F%2Fwww.geocities.co.jp%2FPlaytown-Denei%2F2837%2FLRC.htm>.
- Knihovnu pro Lagarith lze stáhnout na stránkách
<http://lags.leetcode.net/codec.html>, kde jsem ji stáhl také já. Poslední aktualizace programu byla provedena 11. Května 2011 jak uvádí autor webu.

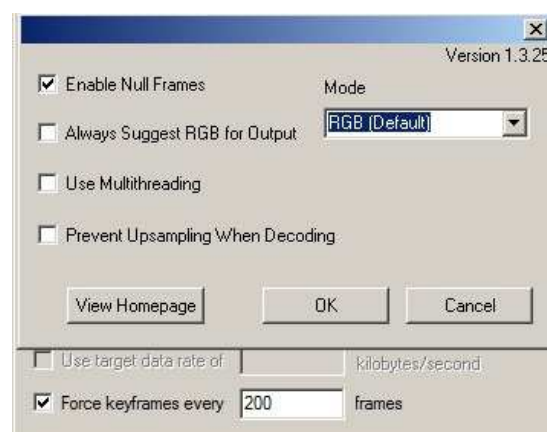
Dále budu už jen uvádět obrázky nastavení jednotlivých metod v programu. Do nějakého složitějšího nastavování jsem se nepouštěl a nechal jsem jednotlivým kódům jejich původní nastavení.



Obrázek 8 - Konfigurace HuffYUV



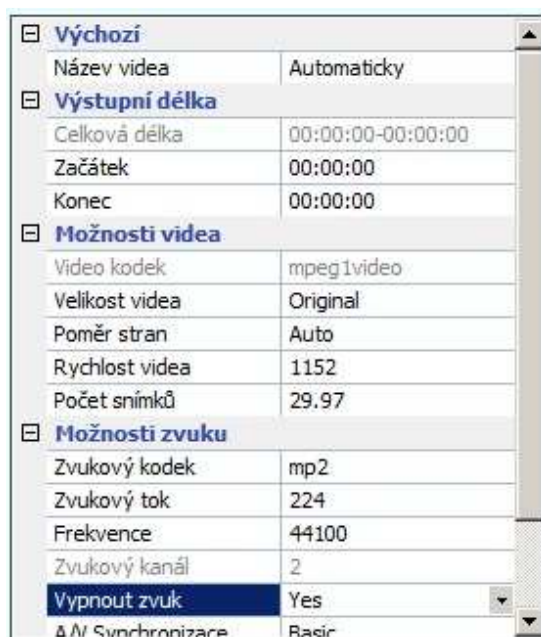
Obrázek 9 – Nastavení AVIzlib



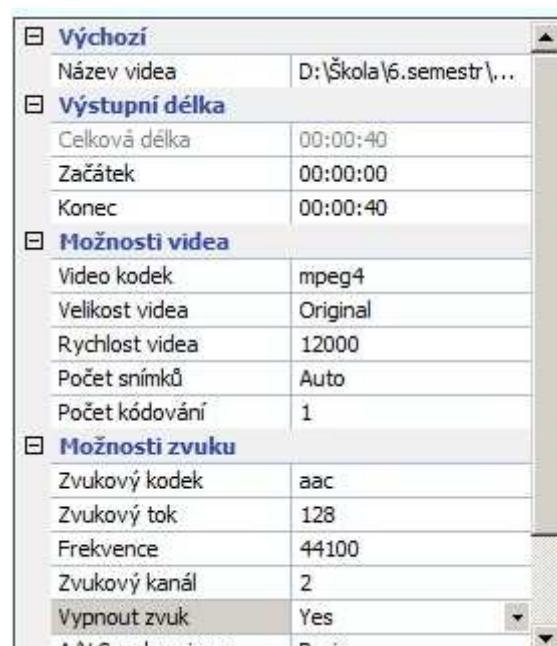
Obrázek 10 – Nastavení Lagarith

3.3.2 Any video convertor

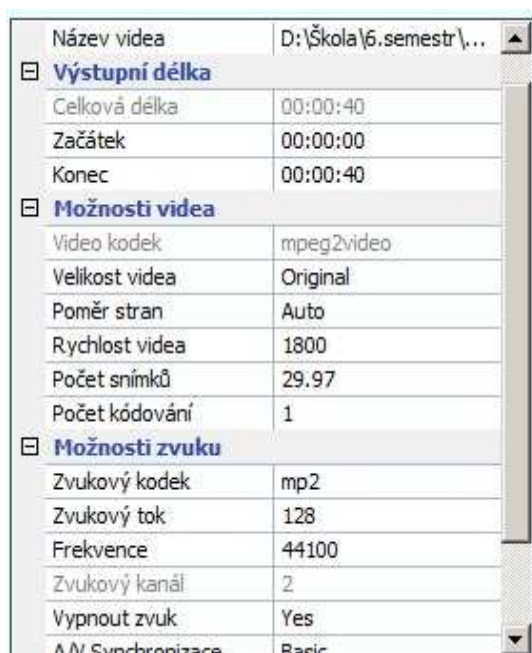
Tento program je rovněž freeware. K dispozici je například na následující stránce: <http://www.edownload.cz/sw/any-video-converter>. Tento program je co se ovládání týče o mnoho jednodušší a má příjemnější grafické prostředí. Ovšem neposkytuje tolik možností nastavení a dalších funkcí jako předchozí program. Nemusíme si do něj jednotlivé standardy stahovat, program už všechno potřebné obsahuje. Stačí tedy pouze do spuštěného programu nahrát video a zvolit jaký výstupní formát chceme, tj. jaké kódování bude použito. Následné obrázky uvádím jako dokumentaci nastavení při mém testování. I v tomto programu jsem většinu nastavení nechal původní, vybral jsem akorát velikost videa tak, aby zůstala původní.



Obrázek 11 – Nastavení MPEG-1



Obrázek 13 – Nastavení MPEG-4



Obrázek 12 – Nastavení MPEG-2



Obrázek 14 – Nastavení WEBM

3.4 Vyhodnocení komprese

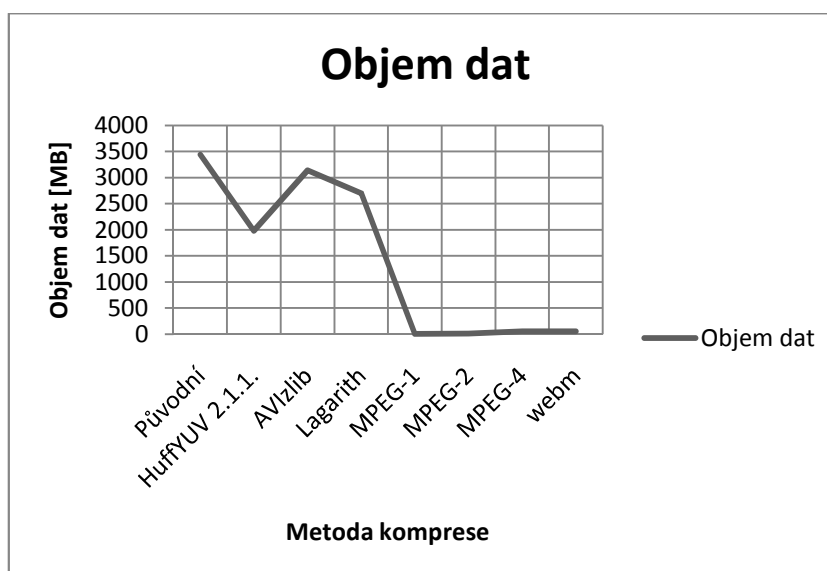
Výslednému videu jsem ponechal výchozí rozlišení (1280x720). Do výsledků jsem přidal ještě standard WEBM, jenž není uveden v teoretické části. Je to jeden z nejnovějších formátů, jehož vývoj je sponzorován společností Google. Jedná se o formát se ztrátovým kódováním a je určen hlavně pro internetová videa. Tento formát využívá populární

sociální síť YouTube. Obrazová kvalita jednotlivých komprimovaných vzorků odpovídá velikostem kompresního poměru nepřímou úměrou. Čím vyšší je kompresní poměr, tím nižší je kvalita videa. To je možné posoudit na jednotlivých vzorcích, které jsou přiloženy k dokumentu. Při porovnání jednotlivých video standardů mohu konstatovat, že novější standardy nám v programu Any Video Converter umožňovaly mnohem větší možnosti nastavení než standardy starší. Toto nastavení ovšem vždy závisí na uživateli podle toho, jaké plány s videem má a s jakou kvalitou obrazu se spokojí. Já jsem se snažil, aby kvalita obrazu zůstala pokud možno nejlepší, a to i na úkor komprese zvláště u ztrátových standardů. K přehrání videí doporučuji program VLC media player, který přehraje bez problému všechny přiložená videa. Nejlépe bych hodnotil ztrátovou komprimaci u standardu webm, tento si zachoval nejlepší kvalitu obrazu. Mezi bezztrátovými standardy za nejlepší považuji HuffYUV, má nejlepší kompresi při stejné obrazové kvalitě.

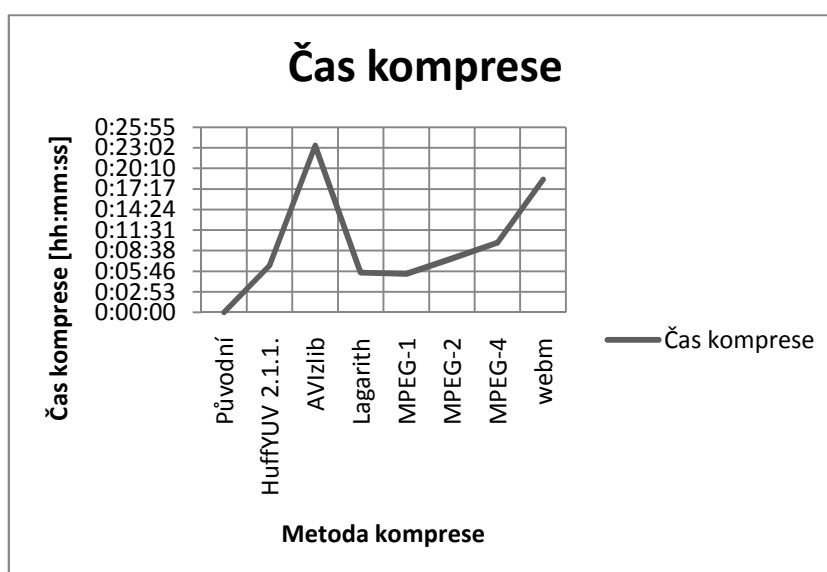
Tabulka 5 – Tabulka výsledků komprimace

metoda (standard)	program	čas komprese [min]	komprimovaná velikost [MB]	Kompresní poměr
HuffYUV 2.1.1.	VirtualDub	6:33	1980	1,74
AVIzlib	VirtualDub	23:21	3140	1,10
Lagarith	VirtualDub	5:34	2700	1,27
MPEG-1	Any Video Converter	5:23	7,14	481,79
MPEG-2	Any Video Converter	7:32	9,73	353,55
MPEG-4	Any Video Converter	9:43	57,30	60,03
webm	Any Video Converter	18:37	57,50	59,83

Graf 1 – Snížení objemu dat



Graf 2 – Časová náročnost kompresí



4 TVORBA WWW STRÁNEK

Stránka byla vytvořena pomocí programu Nvu, který je volně dostupný na stránkách <http://www.mozilla.cz/stahnout/nvu/>. Tento program je sice starší a má již ukončenou podporu. Je nahrazen novějším programem jménem Kompozer, také volně dostupným. Starší program jsem zvolil proto, že obsahuje češtinu a tudíž se mi s ním lépe pracuje. Tyto stránky jsou k dispozici jako příloha k této práci. Zde uvedu pouze náhled stránky.



Obrázek 15 – Náhled stránek

ZÁVĚR

Závěrem si troufám tvrdit, že po přečtení mé práce, pokud by někdo chtěl komprimovat, tak už bude vědět jakým způsobem a pomocí jakých programů, aby byli plně splněny jeho nároky na kompresi, ať už se k ní uchýlil z jakýchkoliv důvodů. Ať to byly důvody uvolnění místa nebo nutnost snížení objemu dat pro lepší přenosovou rychlost. Některé metody nesplnily zcela moje očekávání – např. tvůrce Lagarithu na svých stránkách píše, že je v mnoha ohledech lepší než HuffYUV, což moje testování nepotvrdilo. Může být možné, že jsem někde udělal chybu. Program VisulaDub není tak přehledný a nezkušený uživatel může mít s nastavením problém. Jelikož jsem se do podrobnějšího nastavování nepouštěl, může to být důvod této chyby. Jinak porovnávání ztrátových a bezztrátových metod dopadlo přesně tak, jak dopadnout mělo. Ztrátové metody dosahují mnohokrát vyššího kompresního poměru než metody bezztrátové. Kvalita obrazu se ztrátovou kompresí opravdu sníží, i když u mnou vybraného videa je i ta nejhorší kvalita obrazu MPEG 1 ještě únosná a záznam je zřetelný.

Co se týče www stránek věřím, že mnou nastíněné základy mohou pomoci v rozhodnutí, jaký jazyk a program použít pro vytváření vlastních stránek. Tato oblast je obsáhlá a neustále se vyvíjející a podrobnější popsání této problematiky by vyžadovalo samostatnou práci.

ZÁVĚR V ANGLIČTINĚ

Finally, I hope that after reading my thesis will someone want to compress, will know how to compress, using right programs in order to fulfil all the demands, even though the reasons were different. The reason could be, for instance, the need to reduce the amount of data to increase transmission speed. Some of the methods didn't meet my expectations. For example, the author of Lagarith method says that his method is better in many aspects than HuffYUV but after my testing I cannot confirm it. Of course, it is possible that I made some mistake because VisualDub program is not so well arranged and unexperienced user could have a problem to set it up. As I didn't embark on more precise setting, it could be a reason of this mistake. Otherwise, the comparison of loss and lossless methods turn out exactly as it should. Loss methods achieve much higher compression ratio than lossless methods. Of course, image quality decreases when chosen video is even on the lowest quality MPEG 1 is sufficient and nice to watch.

In terms of web pages, I believe that fundamentals which I outlined help in choosing which language or program to use to create own web pages. This branch is so expansive and still developing that for more concrete description I could write another thesis focused on it.

SEZNAM POUŽITÉ LITERATURY

- [1] CASTRO, Elizabeth. HTML, XHTML a CSS {--} Názorný průvodce tvorbou WWW stránek. 1. vydání. Brno: ComputerPress, 2007. 440 s. ISBN 978-80-251-1531-2
- [2] CAPEK, Jan, FABIAN, Peter. Komprimace dat: Principy a praxe. 1. vydání. Brno: Computer Press, 2000. 190 s. ISBN 8072262319
- [3] FURHT, Borko. Encyclopedia of Multimedia. [s.l.] : Springer US, 2006. 983 s. ISBN 9780-387-30038-2.
- [4] LEVICKÝ, Dušan. Multimediálne telekomunikácie. 1. vydání. Košice: Elfa, s.r.o. 2002. 241 s. ISBN 80-89066-58-5
- [5] SAYOOD, Khalid. Introduction to Data Compression. 3rd edition. [s. l.] : Elsevier Inc., 2006. 704 s. ISBN 978-0-12-620862-7.
- [6] ŽÁRA, Jiří, BENEŠ, Bedřich, SOCHOR, Jiří, FELKEL, Petr. Moderní počítačová grafika. 1. vydání. Brno: ComputerPress, 2004. 612 s. ISBN 80-251-0454-0.
- [7] OSHIMA, Kenji. LCL [online]. 2011 [cit. 2011-05-16]. Dostupné z WWW: <http://www.geocities.jp/sandk_project/LRC.htm>.
- [8] Huffvuv. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2011-05-16]. Dostupné z WWW: <<http://en.wikipedia.org/wiki/Huffvuv>>.
- [9] Huffman coding. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2011-05-16]. Dostupné z WWW: <http://en.wikipedia.org/wiki/Huffman_coding>.
- [10] ROELOFS , Greg; ADLER, Mark . Zlib [online]. 19.dubna 2010 [cit. 2011-05-16]. Dostupné z WWW: <<http://www.zlib.net/>>.
- [11] GREENWOOD, Ben. Lagarith Lossless Video Codec [online]. 11. května 2011 [cit. 2011-05-16]. Dostupné z WWW: <<http://lags.leetcode.net/codecs.html>>.
- [12] DEUTSCH, Peter. DEFLATE Compressed Data Format Specification version 1.3 [online]. Květen 2006 [cit. 2011-05-16]. Dostupné z WWW: <<http://tools.ietf.org/html/rfc1951>>.
- [13] JANOVSKEÝ, Dušan. Jak psát web [online]. 2010 [cit. 2011-05-16]. Dostupné z WWW: <<http://www.jakpsatweb.cz/>>. ISSN 1801-0458.

- [14] GRIMMICH, Šimon . Tvorba webu [online]. 2008 [cit. 2011-05-16]. Dostupné z WWW: <<http://www.tvorba-webu.cz/>>.
- [15] H.264. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, [cit. 2011-05-17]. Dostupné z WWW: <http://en.wikipedia.org/wiki/H.264/MPEG-4_AVC>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ATM	Asynchronous Transfer Mode
AVI	Audio Video Interleave
CD ROM	Compact Disc Read Only Memory
CIF	Common Intermediate Format
CSS	Cascading Style Sheets
DCT	Discrete Cosine Transform
GOB	Groups of Blocks
HDTV	High-Definition Television
HTML	Hypertext Markup Language
JPEG	Joint Photographic Experts Group
kb/s	Kilobit za sekundu
LAN	Local Area Network
LCL	Loss-Less Codec Library
LZ77	Lempel - Ziv 77
Mb/s	Megabit za sekundu
MPEG	Motion Picture Experts Group
NTSC	National Television System Committee
PAL	Phase Alternating Lines
PSTN	Public Switched Telephone Network
QCIF	Quarter Common Intermediate Format
RAM	Random-Access Memory
RGB	Red - Green - Blue
RLE	Run Length Encoding
SECAM	Sequential Couleur Avec Memoire
SNR	Signal/Noise Ratio
VLC	Variable Length Coding
WAN	Wide Area Network
WWW	World Wide Web
XHTML	Extensible Hypertext Markup Language

SEZNAM TABULEK A GRAFŮ

Tabulka 1 - Přenosové rychlosti různých typů videa	10
Tabulka 2 - Mezní parametry úrovní v MPEG-2	15
Tabulka 3 - Algoritmy pro profily v MPEG-2	15
Tabulka 4 - Příklady tagů	24
Graf 1 – Snížení objemu dat	33
Graf 2 – Časová náročnost kompresí	33

SEZNAM OBRÁZKŮ

Obrázek 1 - Příklad videosekvence	12
Obrázek 2 - Video formáty a jejich struktura H.261	17
Obrázek 3 - Video formáty a jejich struktura H.263	18
Obrázek 4 - YUV 1:1:1	20
Obrázek 5 - YUV 4:2:2 (2:1:1)	20
Obrázek 6 - YUV 4:1:1	21
Obrázek 7 - YUV 4:2:0	21
Obrázek 8 - Konfigurace HuffYUV	30
Obrázek 9 – Nastavení AVIzlib	30
Obrázek 10 – Nastavení Lagarith	30
Obrázek 11 – Nastavení MPEG-1	31
Obrázek 12 – Nastavení MPEG-2	31
Obrázek 13 – Nastavení MPEG-4	31
Obrázek 14 – Nastavení WEBM	31
Obrázek 15 – Náhled stránek	34

SEZNAM PŘÍLOH

1. index.html
2. bezztratova.html
3. MENU.html.htm
4. prog.html
5. uvod.html
6. uvod.html.htm
7. vysle.html
8. zav.html
9. ztratova.html
10. Nekomprimované video.avi
11. Komprimace AVIzlib.avi
12. Komprimace HuffYUV.avi
13. Komprimace Lagarith.avi
14. Komprimace MPEH-1.mpg
15. Komprimace MPEH-2.mpg
16. Komprimace MPEH-4.mp4
17. Komprimace webm.webm