

**VIŠJA STROKOVNA ŠOLA ACADEMIA,
MARIBOR**

DIPLOMSKO DELO

**UČINKOVITOST STISKANJA TEKSTOVNIH, SLIKOVNIH,
ZVOČNIH IN VIDEO VSEBIN ZA UPORABO V SPLETU**

Kandidat: David Novak
Študent izrednega študija
Številka indeksa: 11190170016
Program: Medijska produkcija
Mentor: univ.dipl.inž. Marko Peršin

Maribor, oktober 2011

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Podpisan David Novak, št. Indeksa 11190170016, sem avtor diplomskega dela z naslovom Učinkovitost stiskanja tekstovnih, slikovnih, zvočnih in video vsebin za uporabo v spletu,

ki sem ga napisal pod mentorstvom
univ.dipl.inž. Marka Peršina.

S svojim podpisom zagotavljam, da:

- je predložena diplomska naloga izključno rezultat mojega dela;
- sem poskrbel, da so dela in mnenja drugih avtorjev, ki jih uporabljam v predloženi nalogi, navedena oz. citirana skladno s pravili Višje strokovne šole Academia;
- se zavedam, da je plagiatorstvo – predstavljanje tujih del oz. misli, kot moje lastne – kaznivo po Zakonu o avtorskih in sorodnih pravicah; UL št. 16/2007; (v nadaljevanju ZASP), prekršek pa podleže tudi ukrepom VSŠ Academia skladno z njenimi pravili;
- skladno z 32. členom ZASP dovoljujem VSŠ Academia objavo diplomske naloge na spletnem portalu šole.

Maribor, oktober 2011

Podpis študenta:

ZAHVALA

Zahvaljujem se mentorju Marku Peršinu za pomoč in vodenje med pisanjem diplomske naloge, gospodu Klavdiju Šanteju za lektoriranje in Žigu Ocvirk, za pomoč pri izdelavi fotografij, na katerih sem opravljal preizkuse in so predstavljene v tej diplomski nalogi.

POVZETEK

Namen te diplomske naloge je raziskati in predstaviti nekaj najpopularnejših metod in datotečnih formatov, ki se danes najpogosteje uporabljajo za stiskanje teksta, slik, zvoka in videa za uporabo v spletu. Masovna raba interneta zahteva, posebno obdelavo teh vsebin, sicer lahko v nasprotnem primeru hitro trpimo zaradi počasnih prenosov, programsko nekompatibilnostjo in raznimi prostorskimi stiskami.

S stiskanjem podatkov je mogoče učinkovito privarčevati na velikosti digitalnih zapisov, ki so sicer preveč obsežni za normalno rabo v spletu. To storimo tako, da poskušamo iz zapisa odstraniti čim več odvečnih (redundantnih) podatkov. S tem pridejo v igro tudi različne metode stiskanja, kompresijski algoritmi ter datotečni formati, katerih neprimerna uporaba lahko izrazito negativno vpliva na vsebino, ki se stiska. Stiskanje lahko poteka v brez izgubnem ali izgubnem načinu, pri čemer je prvi primernejši za arhiviranje in ohranitev prvotne kvalitete zapisa, drugi pa je bolj smiseln za masovno prenašanje preko spleta in učinkovito varčevanje s prostorom.

Zaradi hitrega razvoja informacijskih tehnologij pa se tukaj pojavi vprašanje glede splošne učinkovitosti tovrstnega stiskanja podatkov, saj ima ta tehnologija tudi svoje slabosti. Slabosti, kot so na primer čas stiskanja in razširitve, programske ter strojne zahteve, razne nekompatibilnosti, popačenja v zapisih in omejitve s strani uporabljenih formatov, lahko hitro povzročijo nevšečnosti pri povprečnemu uporabniku spleta. S tem je cilj te diplomske naloge preizkusiti in raziskati, kako učinkovito je sploh tovrstno stiskane podatkov, kakšne so omejitve tovrstnih datotečnih formatov in kje je meja med učinkovitim in neučinkovitim.

Ključne besede: učinkovitost, spletna vsebina, stiskanje podatkov, tekst, slika, zvok, video, kompresiranje, kodiranje, datotečni format, algoritem, kodek

ABSTRACT

The efficiency of compressing text, image, audio and video contents, for use on the web

The purpose of this diploma is to research and present some of the most popular methods and file formats, which are used these days for the compression of text, images, sound and video, to be used on the web. The massive usage of the internet demands a special way of processing these contents, otherwise we can quickly suffer from slow download speeds, program incompatibilities and various space limitations.

With data compression it is possible to efficiently reduce the size of such digital records, which are otherwise too extensive to be normally used on the web. We achieve this by reducing the redundant data as much as possible. As we do this, various compression methods, algorithms and file formats come to use, by which their improper usage can have a negative affect on the content that is being compressed. Such compression can be done in a lossless or lossy manner, by which the first one is suitable for archiving and the preservation of the primary record quality, and the other is more suitable for massive data transfers and efficient savings of space.

Because of the rapid development of information technologies, a question of the overall efficiency of such data compression occurs, because this technology has its weaknesses. Weaknesses, such as the time needed for the compressing and decompressing data, software and hardware requirements, various incompatibilities, record distortions and format limitations, can quickly become unpleasant for the average web user. That is why the goal of this diploma is to test and research, how efficient such data compression actually is, what are the limitations of such file formats and where is the border between efficient and inefficient.

Keywords: efficiency, web content, data compression, text, image, sound, video, compression, coding, file format, algorithm, codec

KAZALO VSEBINE

Kazalo tabel	8
Kazalo slik	9
Seznam kratic in izrazov	10
1 Uvod	12
1.1 Opredelitev obravnavane zadeve	12
1.2 Namen, cilji in osnovne trditve	12
1.3 Predpostavke in omejitve	13
1.4 Opomba bralcu	13
1.5 Metode testiranja	14
2 Spletna vsebina	15
2.1 Pretočne vsebine	16
2.2 Piratska vsebina	17
3 Pretvorba analognega zapisa v digitalnega	18
3.1 Shranjevanje podatkov	18
3.1.1 Podatkovni tok	18
3.1.2 Datoteka	18
3.1.3 Format datoteke	19
3.1.4 Kontejner format	20
3.1.5 Končnica datoteke	20
4 Stiskanje podatkov	22
4.1 Brez izgubno stiskanje	23
4.2 Izgubno stiskanje	23
4.3 Metode stiskanja	24
4.4 Kompresijski algoritmi	25
4.5 Programi za stiskanje datotek	26
5 Učinkovitost stiskanja podatkov	29
5.1 Merjenje učinkovitosti	30
6 Tekst v spletu in njegove lastnosti	31
6.1 HTTP kompresija	33
6.2 Stiskanje tekstovnih datotek	34
6.3 Razširjeni arhivsko-kompresijski formati	34
6.3.1 ZIP	34
6.3.2 RAR	35

6.3.3	7z	35
6.4	Učinkovitost stiskanja tekstovnih datotek	36
6.5	Ugotovitve	38
7	Slike v spletu in njihove lastnosti	40
7.1	Rastrska slika	40
7.2	Vektorske slike	41
7.3	Stiskanje in optimiziranje slik.....	42
7.4	Razširjeni grafični formati	45
7.4.1	JPEG	45
7.4.2	GIF.....	46
7.4.3	PNG	46
7.4.4	SVG	47
7.5	Učinkovitost stiskanja slikovnih datotek	47
7.6	Ugotovitve	50
8	Zvok v spletu in njegove lastnosti	52
8.1	Stiskanje zvoka	55
8.1.1	Stiskanje govora	58
8.2	Razširjeni zvočni formati.....	59
8.2.1	MP3	59
8.2.2	AAC.....	59
8.2.3	WMA.....	60
8.2.4	Vorbis	60
8.2.5	FLAC	61
8.3	Učinkovitost stiskanja zvočnih datotek	61
8.4	Ugotovitve	64
9	Video v spletu in njegove lastnosti.....	66
9.1	Video v računalništvu	67
9.2	HTML5	70
9.3	MPEG-4.....	70
9.4	Stiskanje videa	70
9.5	Razširjeni video formati.....	74
9.5.1	H.264	75
9.5.2	Theora.....	75
9.5.3	VP8.....	75
9.5.4	Xvid	76

9.5.5	VC-1 / Windows Media Video 9	76
9.6	Razširjeni kontejner formati	77
9.7	Učinkovitost stiskanja video datotek	78
9.8	Ugotovitve	82
10	Zaključek in ugotovitve	84
11	Viri in literatura	86
11.1	Knjige	Napaka! Zaznamek ni definiran.
11.2	Spletni viri	86

KAZALO TABEL

Tabela 1:	Preizkus stiskanja TXT, DOC in PDF tekstovnih formatov	37
Tabela 2:	Stiskanje različne tekstovne vsebine s programom WinRAR	38
Tabela 3:	Stiskanje barvne slike	49
Tabela 4:	Stiskanje sivinske slike	49
Tabela 5:	Stiskanje črno-bele slike	49
Tabela 6:	Stiskanje barvne slike nizke ločljivosti	50
Tabela 7:	Stiskanje SVG (Scalable Vector Graphic) datotek	50
Tabela 8:	Stiskanje v brez izgubnem FLAC formatu	62
Tabela 9:	Preizkus stiskanja s formati MP3, AAC, WMA in Vorbis	62
Tabela 10:	Stiskanje zvočnih datotek v VBR načinu	63
Tabela 11:	Stiskanje govora pri 16 Kbps (širokopasovne povezave)	64
Tabela 12:	Najpogostejša razmerja in ločljivosti zaslonov digitalnih naprav	69
Tabela 13:	Tabela primerjave kontejner formatov	77
Tabela 14:	Stiskanje videa visoke kakovosti: 1024 × 576, bitna hitrost: 5 Mbps	81
Tabela 15:	Stiskanje videa srednje kakovosti: ločljivost: 640 × 360, bitna hitrost: 2,5 Mbps	81
Tabela 16:	Stiskanje videa nizke kakovosti: ločljivost: 356 × 200, bitna hitrost: 600 Kbps ...	82

KAZALO SLIK

Slika 1: Primer prirejanja slike, ki bo uporabljena v spletu	15
Slika 2: YouTube omogoča pretakanje tudi celotnih oddaj na zahtevo ali v živo	17
Slika 3: Primer datotečnega kontejner formata z enim zvočnim in enim video tokom	20
Slika 4: Primer slike formata JPEG, ki je shranjena v datoteki.....	21
Slika 5: Princip delovanja kodirnika in dekodirnika (kodeka)	27
Slika 6: Struktura binarne tekstovne datoteke	31
Slika 7: Rastrska slika s primerom izredne povečave	41
Slika 8: Primer vektorsko narisane avtomobila	42
Slika 9: Iz različnih barvnih pik sestavljena črta.....	43
Slika 10: Izvirna slika na levo in slika stisnjena v JPEG formatu na desno.....	44
Slika 11: Posledice vedno bolj intenzivnega stiskanja iz leve proti desni.....	45
Slika 12: Stiskanje barvne, sivinske in črno-bele slike	48
Slika 13: Ponazoritev zvočnega valovanja skozi zrak.....	52
Slika 14: Groba ponazoritev pretvorbe iz analognega zvočnega zapisa v digitalnega.....	54
Slika 15: Prikaz digitalnega zvočnega zapisa pri različnih stopnjah povečave.....	54
Slika 16: Programsko predstavljeno valovanje zvoka in njegove potencialne možnosti za brez izgubno stiskanje	57
Slika 17: Primer frekvenčnega maskiranja.....	58
Slika 18: Video kot kombinacija zaporednih slik in zvoka	66
Slika 19: Primer kompenzacije gibanja	73
Slika 20: Primer "podvzorčenja" pri različnih nivojih	74
Slika 21: Video posnetek dinamičnih kadrov na levi in statičnih na desni	79

SEZNAM KRATIC IN IZRAZOV

Barvna pika – Najmanjša enota digitalne slike.

Barvni model – Matematični način predstavljanja barv, kot kombinacije več številčnih vrednosti.

Bit – Osnovna enota informacije, ki se podaja v računalništvu in telekomunikacijah.

Bitna hitrost – Število bitov, ki so preneseni ali procesirani v nekem časovnem obdobju (načeloma v eni sekundi).

Datoteka – Osnovna oblika hranjenja digitalnih podatkov.

Dekodirnik – Program (ali algoritem), ki dekodira podatke.

Format datoteke – Način, s katerim so podatki zapisani v datoteko.

ISO (International Standards Organization) – Organizacija, ki skrbi za razvoj standardov na področju informacijskih tehnologij in računalništva, med drugimi tudi za razvoj JPEG in MPEG kompresijskih standardov.

Koda – Simbol, ki predstavlja nek drug simbol ali zlog znakov.

Kodek – Izraz, ki predstavlja kodirnik in dekodirnik hkrati.

Kodirnik – Program (ali algoritem), ki kodira podatke.

Kompresija (digitalna) – procesiranje ali stanje digitalnih podatkov, ki so bili obdelani tako, da zavzamejo manj prostora, kakor so ga zasedli prvotno.

Ločljivost – Število pik, ki sestavljajo sliko in jih je zaslon sposoben prikazati.

Mono – Izraz, ki se uporablja v avdio reprodukciji za predstavitev enega neodvisnega zvočnega kanala (sledi).

MPEG (Moving Picture Experts Group) – skupina strokovnjakov, ki skrbi za razvoj standardov pri prenosu in stiskanju zvoka in videa.

Podatkovni tok – Podatki, ki se nahajajo v pomnilniku in se prenašajo ali obdelujejo.

Pretakanje – Metoda konstantnega prenašanja digitalnih podatkov od strežnika k prejemniku v realnem času.

Simbol – najmanjša enota podatka, ki se lahko stisne.

Stereo – Izraz, ki se uporablja v avdio reprodukciji za predstavitev dveh neodvisnih zvočnih kanalov (sledi).

Strežnik – Računalnik, ki izvaja neko storitev v omrežju, kot je prenos vsebine, nadzor povezave, zaščita in podobno.

1 UVOD

1.1 *Opredelitev obravnavane zadeve*

Z bliskovito rastjo svetovnega spleta, raste tudi bogatost njegove vsebine. Danes se poleg surovega teksta in slik, na spletu vse pogosteje pojavljajo tudi raznorazni zvočni in video zapisi, kateri lahko hitro napolnijo diske strežnikov in močno obremenijo promet v omrežju. Kljub konstantnim nadgradnjam omrežij pa se je izkazalo, da samo to ni dovolj, zato se strokovnjaki lotevajo problema že pri samem izvoru podatkov in to tako, da vso to obsežno vsebino stisnejo. Ideja o stiskanju podatkov je že relativno stara in je v uporabi že kar nekaj časa, vendar se je ta zaradi večanja števila in velikosti podatkov, ki se danes prenašajo po internetu in njihove splošne rabe v informatiki, izkazala, kot nujno potrebna. S stiskanjem podatkov lahko privarčujemo na kapaciteti strežniških diskov in še pomembnejše, na hitrosti prenosa podatkov, po drugi strani pa je potrebno tovrstne podatke na koncu tudi razširiti, če jih želimo uporabljati. Pri tem se srečamo z raznoraznimi programi za delo s tovrstnimi podatki ter datotečnimi formati, ki so zaradi njihove računalniške in multimedijske narave, pritegnili tudi mojo pozornost, saj se z njimi ogromno srečujem.

1.2 *Namen, cilji in osnovne trditve*

Stiskanje ali bolj strokovno kompresiranje ali kodiranje podatkov, je proces, ki obdela podatke tako, da ti po obdelavi zasedejo manj prostora, kakor so ga prvotno. Gre torej za pretvarjanje informacij z namenom varčevanja, kakor včasih, kadar nastavimo tiskalnik na varčen način tiskanja z namenom, da bi privarčevali na barvi. Tovrstno stiskanje podatkov se danes uporablja že pri vseh sodobnih informacijskih tehnologijah, pri spletni vsebini, kakor so tekst, slike, zvok in video, pa želimo s stiskanjem pridobiti predvsem na velikosti njihovih datotek s čimer pridobimo tudi na hitrosti prenosa. Istočasno moramo tukaj upoštevati standarde, s katerim zagotovimo uporabo teh datotek na vsakem računalniku, vsaki mobilni napravi in celo na televizorju.

Večina vsebine je torej stisnjena, ampak obsežnost svetovnega spleta in informacijskih tehnologij nas lahko hitro preplavi z raznoraznimi datotečnimi formati in programi, ki nas lahko hitro zmedejo. Namen tega diplomskega dela je raziskati lastnosti omenjenih spletnih

vsebin in kako se te stiskajo, ob tem pa raziskati najpopularnejše datotečne formate, ki se tukaj uporabljajo in ugotoviti njihove lastnosti, prednosti in slabosti.

Z uporabo tovrstne tehnologije stiskanja pa lahko hitro naletimo tudi na njene slabosti. Med brskanjem po spletu lahko večkrat naletimo na izredno nekvalitetno vsebino, ki je zelo verjetno, z izjemo nekvalitetnega izvora, posledica slabega stiskanja podatkov. Prav tako se hitro zgodi, da si nekatere vsebine ne moremo ogledati ali odpreti, kar je ponovno lahko posledica neprimerne stiskanja podatkov. Ob takšnih in drugačnih težavah se tako vprašamo, kako učinkovito je sploh takšno stiskanje podatkov, glede na to, da lahko povzroči precej preglavic. Cilj te diplomske naloge je tako raziskati, kolikšen je sploh prihranek pri stiskanju datotek in kakšne so posledice, ki pri tem nastanejo. Tukaj želim z uporabo primernih programov narediti preizkuse, ki bodo pokazali učinkovitost teh formatov, z ugotovitvami pa bom lahko tudi sam uspešno stiskal različne podatke in pri tem uporabljal primerne formate, če se bo tovrstno stiskanje podatkov izkazalo kot učinkovito.

1.3 Predpostavke in omejitve

Danes so tehnologije stiskanja podatkov že izredno kompleksne in razširjene. Njihov razvoj zahteva ogromno znanja s področja računalništva, informatike, matematike in informacijske teorije. Literature s tega področja so izredno obsežna, zato se bom skozi diplomsko nalogo poskušal osredotočiti samo na metode, ki se uporabljajo predvsem v spletu, njihova uporaba pa je tudi najbolj razširjena. Navajal bom tudi angleške izraze, saj tovrstnega gradiva, oziroma literature, pri nas ni veliko, raba slovenskih izrazov na tem področju pa je za enkrat še skopa.

1.4 Opomba bralcu

Obsežen naslov diplomske naloge zajema prav tako obsežno področje iz računalništva in informacijskih tehnologij, zato so navedeni primeri in preizkusi, ki sledijo, bolj splošne narave. Njena struktura je sestavljena iz uvodnih poglavij, ki predstavijo področje spletnih vsebin in osnovne principe stiskanja podatkov, sledijo pa jim poglavja, ki natančneje predstavijo njihove lastnosti, načine stiskanja in učinkovitost formatov s področja teksta, slike, zvoka in videa. Vsako tovrstno poglavje ima podane vse programe, ki so bili uporabljeni za izvajanje testov ter ugotovitve na koncu vsakega poglavja, s katerimi si lahko bralec pomaga ob izdelavi spletnih strani ali pripravi optimalne spletne vsebine.

1.5 Metode testiranja

Testirani bodo samo relevantni datotečni formati in to s privzetimi nastavitvami ter nastavitvami, ki se najpogosteje uporabljajo pri uporabi tovrstne vsebine v spletu. S tem bo mogoče dobiti posplošeno predstavo o njihovi učinkovitosti, ne pa fiksnih rezultatov, saj je njihov proces stiskanja močno odvisen od lastnosti vsebine, ki se stiska, programske opreme in podanih parametrov. S stiskanjem se bom večinoma osredotočal na ohranitev čim višje kvalitete zapisa. Za lažje razumevanje bom skozi poglavja vedno poskušal prikazati dva teoretična primera iz stiskanja podane vsebine, sledila pa bo serija praktičnih preizkusov, kjer bom prikazal rezultate in podal ugotovitve ter predloge.

Preizkuse bom izvajal na računalniku s sledečimi specifikacijami:

- Intel Core 2 Duo E8400 (3,0 GHz)
- 4 GB RAM (DDR2, 800 MHz)
- Gigabyte EP45-DS3R
- NVIDIA GeForce 9800 GT (512 MB)
- Microsoft Windows 7 Ultimate SP1, 64 bit

Poleg računalnika bom za zajemanje in izdelavo vsebine uporabil sledeče naprave:

- Fotoaparat Canon EOS 50D
- Mikrofon Vivanco DM10
- Videokamera Sony DCR-TRV60E PAL

Rezultati (velikosti stisnjenih datotek) so podani po internacionalnem sistemu enot:

- 1 KB (**kilobajt**) = 1024 B (bajtov) za *meritev velikosti datoteke*
- 1 kb (**kilobit**) = 1000 b (bitov) za *meritev hitrosti prenosa podatkov (bitne hitrosti)*
- 1 B (**bajt**) = 8 b (bitov)
- 1 B/s = 8 bps za *prikaz podatkovne hitrosti ali prenosa podatkov*
- 1 GB = 1024 MB = 1.048.576 KB = 1.073.741.824 B
- 1Gb = 1000 Mb = 1.000.000 Kb = 1.000.000.000 b

2 SPLETNA VSEBINA

Spletno vsebino predstavlja vsak tekst, slika, zvok, video, animacija, spletna aplikacija, dokument, storitev ter vse ostalo, kar se pojavi na naših zaslonih kadar brskamo po svetovnem spletu oziroma internetu. Ker je namenjena širši javnosti in dostopna na globalni ravni, mora ta biti prilagojena tako, da jo lahko odpira in obdeluje večina programske in strojne opreme, ki ima dostop do svetovnega spleta. Gre torej za vso tekstovno, vizualno ali slušno vsebino, ki jo z brskanjem po spletu, prenašamo na svoj osebni računalnik ali kakšno drugo multimedijско napravo, ki zna prikazovati spletno vsebino.

V tej diplomski nalogi sem se osredotočil predvsem na tekstovno, slikovno, zvočno in video vsebino. Ti elementi spadajo med najpopularnejše gradnike spletne vsebine in so zaradi tega tudi zelo natančno prilagojeni za uporabo v spletu, saj morajo zasedati čim manj prostora, držati se standardov, ter po kakovosti zadovoljiti povprečnega uporabnika spleta. S tem je na primer slika, ki je bila zajeta s profesionalnim studijskim fotoaparatom, po lastnostih drugačna, kot na primer ta ista slika, ki se kasneje pojavi v spletu. Slika na spletu je najverjetneje trikrat pomanjšana in pretvorjena v nek zelo standardiziran in razširjen format. Prvotno sliko lahko sicer prav tako uporabimo v spletu, vendar bi bila ta zaradi svoje velikosti in formata, skrajno neprimerna za tovrstno uporabo. Podobna zgodba je z uporabo teksta, zvoka in videa.



Slika 1: Primer prirejanja slike, ki bo uporabljena v spletu

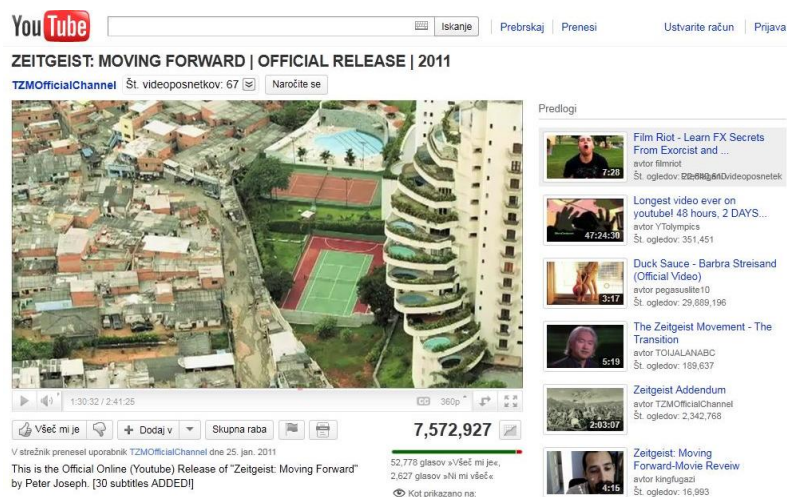
Vir: <http://www.openclipart.org/detail/6682/dslr-camera-by-flomar>, dne 4.3.2011,
<http://www.openclipart.org/detail/137107/guepard-on-landscape-by-gustavorezende>, dne 4.3.2011 in <http://www.openclipart.org/detail/140437/empty-monitor-by-easy>, dne 4.3.2011,
zajeta zaslonska slika programa ACDSee Photo Manager 12; lasten

Za strokovnjake na področju informacijskih tehnologij, je danes že samoumevno, da se vso pridobljeno gradivo še naknadno obdela in priredi tako, da kasneje čim bolj služi svojem namenu, bodisi za domačo uporabo, arhiviranje, profesionalne namene ali uporabo v spletu.

2.1 *Pretočne vsebine*

Medtem, ko sta tekst in slika že neke vrste osnovna gradnika spletnih strani, je uporaba zvoka in videa šele v zadnjih nekaj letih, postala izredno popularna. Zvok in video sta nam bistveno bolj naravna za komunikacijo, zato njuna uporaba danes iz dneva v dan raste. V nasprotju s tekstom in sliko pa slednja zahtevata bistveno več podatkov, za njun zapis v digitalni obliki, kar posledično pomeni tudi zahtevnejši prenos preko interneta. Pred leti je bil prenos zvoka in videa za domače uporabnike prava redkost. Okoli leta 2000 pa smo bili priča razcvetu širokopasovnih povezav, širši uporabi standardov in formatov ter splošni komercializaciji svetovnega spleta, s čimer je njun prenos preko spleta postal tudi smiseln. Tovrstne prenose so večinoma spremljali dve slabosti: dolgotrajen prenos in nepodprtost formata. Torej, ne samo, da je bilo potrebno dolgo čakati na prenos datoteke, po končanem prenosu je bila tudi velika verjetnost, da njene vsebine nismo morali predvajati.

S pretakanjem (angl. streaming) zvoka ali videa pa je možno do te vsebine dostopati takoj in si jo ogledati ali poslušati, v realnem času. Ta tehnologija, ki je sorodna televizijskemu načinu oddajanja signala, je kmalu pomenila veliko spremembo v spletu in je še danes v razcvetu. Danes so pretočne vsebine zaradi hitrih povezav, zmogljivih naprav in enostavnosti uporabe, postale prvoten način za prenos zvoka in videa. Spletne strani, kot so YouTube ali SHOUTcast, so danes masovno obiskane, saj nam nudijo ogromno količino pretočnih videov in radijskih postaj, katere si lahko ogledamo ali poslušamo brez prenosa datotek, na zahtevo ali v živo. Prenos vsebine se tukaj izvaja v ozadju, v obliki tokovnega pretakanja podatkov. V igri pa so prav tako razni formati in algoritmi za takojšnje dekodiranje in predvajanje te vsebine. (http://en.wikipedia.org/wiki/Streaming_media, dne 3.3.2011)



Slika 2: YouTube omogoča pretakanje tudi celotnih oddaj na zahtevo ali v živo

Vir: <http://www.youtube.com/watch?v=4Z9WVZddH9w>, dne 5.3.2011

Klasičen način prenosa podatkov se sicer še vedno uporablja, vendar je uporaba pretočnih vsebin postala izredno popularna.

2.2 Piratska vsebina

Po internetu masivno kroži še ena vrsta spletne vsebine, ki se načeloma ne bi smela, saj je njena distribucija in prenos, ilegalno početje. Gre za tako imenovano piratsko vsebino, ki omogoča brezplačen prenos in uporabo sicer plačljive in avtorsko zaščitene vsebine. Vsebinsko gledano gre tukaj predvsem za filme, glasbo in programsko opremo, katero je mogoče prenesti s pomočjo raznih programov, omrežij ali pa jo enostavno prenesemo s strežnikov, ki omogočajo brezplačno gostovanje datotek.

Zanimiva plat tovrstne vsebine je ta, da se razvija v korak s tehnologijami omrežja in medijev. Tukaj se pojavlja vprašanje ali bi vsi današnji predvajalniki, mobilni telefoni in televizorji, vseeno znali predvajati popularne formate, kot so MP3 in DivX/Xvid, če ti ne bi bili tako priljubljeni med izdajatelji piratskih vsebin? Kakorkoli že, vredno je omeniti, da je tehnologija stiskanja podatkov, bistvenega pomena tudi za pripravljavce piratske vsebine. Ti se namreč zaradi obsežnih prenosov trudijo čim bolj učinkovito stisniti vsebino, kar pa posledično privede do neke vrste testiranja uporabljenih formatov in parametrov za stiskanje. Piratska vsebina nam je danes na voljo tudi v visokih ločljivostih (HD) in se po spletu prenaša v novejših formatih, kot so H.264 ali DivX Pro ter v obliki pretočnih vsebin.

3 PRETVORBA ANALOGNEGA ZAPISA V DIGITALNEGA

Internet je izključno digitalni sistem, zato je vsa njegova vsebina prav tako digitalno kreirana, z uporabo programov ali pa je analogno zajeta z napravami (na primer fotoaparatom, mikrofonom, video kamero...), ki nato zajeto vsebino pretvorijo v digitalno obliko. Ta proces pretvarjanja analognega zapisa v digitalnega imenujemo digitalizacija in predstavlja proces, brez katerega ne gre.

Z digitalizacijo analognega zapisa (signala) želimo prvotno valovanje zvoka in odbojev svetlobe, katere zajemamo kot analogen signal, ponazoriti z binarno kodo - številom 0 in 1. Narava pa seveda ne deluje na principu ničel in enic, zato je digitalizacija v praktičnem smislu, samo približek signala, ki ga predstavlja. Digitalizacija ima s tem tudi svoje slabosti. Prva od teh, je izguba informacij, ki se pojavi med pretvarjanjem; druga slabost pa je velikost pretvorjenega zapisa v digitalni obliki. Natančno pretvarjanje analognega signala v digitalnega namreč tvori ogromno količino izhodnih bitov (ničel in enic), katerega rezultat je velika datoteka, zato se tovrstne podatke (načeloma po obdelavi) vedno stiska. (<http://en.wikipedia.org/wiki/Digitizing>, dne 5.4.2011)

3.1 Shranjevanje podatkov

Kadar se podamo v svet interneta, sprožimo prenos informacij med računalniki v obliki paketov. Ti paketi kasneje na našem računalniku skupaj tvorijo datoteko ali tok podatkov. Vsa zajeta ali programsko kreirana vsebina, z izjemo pretočnih vsebin, je tako ne glede na vrsto podatkov v kateri je podana ali so ti stisnjeni ali ne, shranjena v obliki računalniških datotek.

3.1.1 Podatkovni tok

Podatke ni potrebno vedno zapisati in shraniti v datoteko, temveč jih lahko v obliki podatkovnega toka (angl. data stream), direktno prenašamo obdelujemo. Podatkovni tok je torej tok podatkov, ki se pretaka med pošiljateljem in prejemnikom ali pa predstavlja tok podatkov, ki je shranjen v datoteki.

3.1.2 Datoteka

Računalniška datoteka je zaporedje bitov, ki so zaokroženi v celoto in so tako na voljo uporabniku, kot samemu sistemu. Ta predstavlja osnovno obliko hranjenja digitalnih podatkov, zato se prav tako vsa spletna vsebina shrani v datoteke. Podatki znotraj datoteke se lahko

med seboj razlikujejo. Na grobo lahko govorimo o binarnih in ASCII datotekah, vsebinsko pa lahko ti zapisi predstavljajo vse od zvočnega zapisa do programov, zato se glede na vrsto in način zapisa v datoteki, datoteke ločijo po njihovem formatu.

3.1.3 *Format datoteke*

Format datoteke ali datotečni format, nam pove na kakšen način so podatki v datoteki zapisani oziroma na kakšen način so črke, barvne pike ali zvok predstavljeni. S tem mora program, ki bo uporabljal te podatke v datoteki, tudi prepoznati, sicer mu bodo ti podatki neznani in neuporabni. Vse tekstovne, slikovne, zvočne in video vsebine so prav tako shranjene v svojih datotečnih formatih. Te lahko odpiramo z različnimi programi, ki razumejo (podpirajo) te formate. Spletni brskalnik na primer, zna med drugimi procesirati in prikazovati datoteke formata HTML, ne zna pa predvajati datoteko formata MP3. Zvočno datoteko formata MP3 zato odpremo s programom, ki ima v sebi že vgrajen dekodirnik za branje zvočnega zapisa, shranjenega v tem datotečnem formatu.

Ker stiskanje podatkov pomeni pretvarjanje podatkov v drugo obliko zapisa, je datotečni format tisti, ki predstavlja tisto drugo (stisnjeno) obliko teh podatkov. Veliko datotečnih formatov podpira oz. shranjuje podatke v že stisnjeni obliki, pri čemer se stiskanje izvede že med samim shranjevanjem podatkovnega toka v datoteko. S tem je učinkovitost stiskanja povezana tudi z samo sposobnostjo tega datotečnega formata.

Kljub temu, da vsi datotečni formati ne podpirajo stiskanja podatkov, je tovrstnih formatov veliko, celo tako veliko, da včasih med seboj tekmujejo za prevlado. Različne formate razvijajo različna podjetja in organizacije, kar pomeni, da so ti tudi njihova lastnina in lahko njihovo uporabo omejijo. Večja podjetja, kot so na primer Apple, Adobe ali Microsoft, razvijajo formate, ki so večinoma namenjeni izključno za uporabo z njihovimi programskimi okolji. To so tako imenovani zaprti formati, poznani tudi kot lastniški formati, katerih raba se plačuje z nakupom njihove licence. Obstajajo pa tudi tako imenovani odprti formati, katerih uporaba je brezplačna in namenjena za vse, in formati, ki veljajo za internacionalne standarde.

Datotečne formate, lahko delimo v tri družine:

- Internacionalni (ISO, ITU, MPEG, JPEG...)
- Lastniški (Apple, Microsoft, DivX Inc., RealNetworks...)
- Odprti (Xiph.Org Foundation, Open Source Initiative...)

3.1.4 Kontejner format

V primeru zvočnih in video vsebin, se večinoma srečamo s posebno vrsto datotečnega formata imenovan kontejner (angl. container). Poznan tudi kot paketni (angl. wrapper) format, je datotečni format, ki se obnaša kot shramba, v katero lahko shranimo enega ali več podatkovnih tokov (formatov). V primeru navadne video datoteke, bi takšen kontejner najverjetneje vseboval en podatkovni tok videa (brez zvoka) in en podatkovni tok zvoka (brez videa), oba zapisana v svojem formatu. Te načeloma spremljajo še meta podatki (XML, RDF, XMP, ...) o podatkovnih tokih in vsebini. Predvajalnik, ki zna odpreti takšen kontejner, tako najprej analizira podatkovne toke v kontejnerju, nato pa jih poskuša sinhrono dekodirati in predvajati. Na podobnem principu delujejo tudi arhivsko-kompresijski formati, katere bomo spoznali malo kasneje. Vsi datotečni formati niso združljivi z vsemi vrstami kontejnerjev. (<http://www.reelseo.com/file-formats-containers-compression/>, dne 17.3.2011)

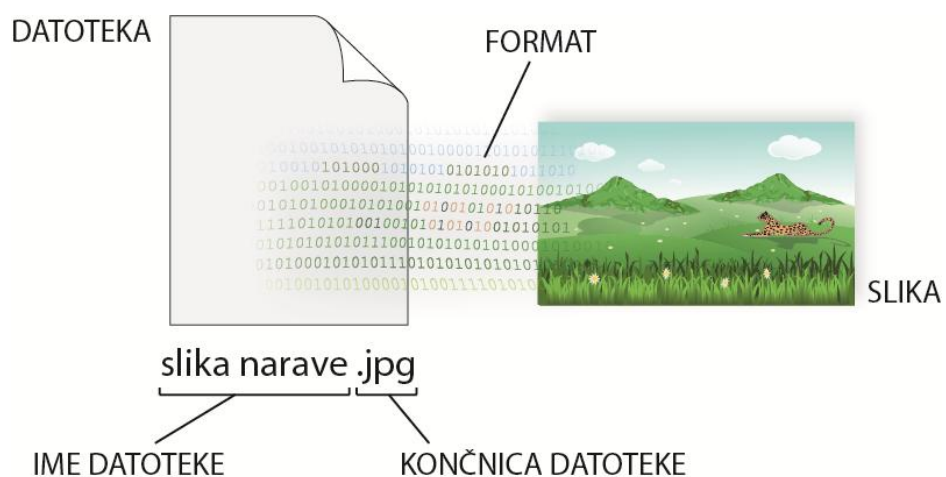


Slika 3: Primer datotečnega kontejner formata z enim zvočnim in enim video tokom

Vir: Kill The Irishman, 2011; lasten

3.1.5 Končnica datoteke

Glede na format, ki ga vsebuje datoteka, se datoteki določi temu primerna končnica. Končnica datoteke je zelo razširjena metoda, ki jo uporablja večino operacijskih sistemov za lažje prepoznavanje vrste formata v datoteki. Prepoznamo jo po črkah in številkah, ki se nahajajo za ločilno piko v imenu datoteke (npr. ime_datoteke.končnica) in je večinoma predstavljena s tremi znaki, kot na primer: .jpg, .mp3, .doc, .zip,...



Slika 4: Primer slike formata JPEG, ki je shranjena v datoteki

Vir: <http://www.openclipart.org/detail/137107/guepard-on-landscape-by-gustavorezende>, dne 5.3.2011; lasten

4 STISKANJE PODATKOV

Stiskanje, kompresiranje ali izvirno kodiranje podatkov, je proces, ki pretvori voden tok podatkov (iz samega izvora ali prvotno shranjenih podatkov) v drug tok podatkov, ki je po velikosti manjši (manj redundanten) od prvega. Takšen tok podatkov je lahko datoteka ali podatki, ki se nahajajo v pomnilniku in jih lahko nato še programsko obdelamo (stisnemo). (Salomon, Motta, 2010, 2)

Tovrstni programi storijo to tako, da vsebino datoteke ali tok podatkov sprejema in procesira, vse dokler ne zazna konca podatkov. Med procesom je tako nastal nov zapis podatkov, ki sedaj lahko pomeni nov tip datoteke ali pa spremenjeno staro datoteko. Procesirani podatki so lahko znaki, slikovne pike ali zvočni zapisi in v večini primerov, kodirnem programu ne predstavljajo pomena. Ti se najprej preberejo in analizirajo, nato pa po določenem algoritmu obdelajo, pri čemer se isti algoritem kasneje uporabi tudi za odpiranje teh obdelanih (stisnjenih) podatkov. Novo nastala datoteka ali podatkovni tok, lahko sedaj vsebuje kodirno shemo (slovar), ki služi kot neke vrste navodilo, kako to datoteko ali tok, razširiti.

Vsako tovrstno stiskanje temelji na ideji odstranjevanja redundantnih podatkov, torej, iz pretvarjanja neefektivnega (dolgega in obsežnega) v efektivno (krajše in kompaktno), kar lahko pomeni tudi pretvorbo datotečnega formata. Tehnologija stiskanja podatkov je že relativno stara. Teoretične razlage s tega področja se prvič pojavijo konec štiridesetih in v začetku petdesetih. V svetu informacijskih tehnologij se zaradi izrednih količin podatkov, danes izredno veliko uporablja, še posebej pa je prišla do izraza v zadnjih nekaj letih s komercializacijo svetovnega spleta in razvojem multimedijskih naprav. V primeru spletne vsebine je stiskanje podatkov aktualno predvsem zaradi dveh razlogov:

- **hitrejši prenos** (manjša količina podatkov pomeni krajši čas prenosa);
- **bolj učinkovito shranjevanje** (manjše datoteke zasedejo manj prostora, zato jih lahko na shranjujemo v večjih količinah).

Kljub prednostim, ki jih pridobimo s stiskanjem pa le ne gre brez slabosti. Vse podatke, ki stisnemo, je kasneje potrebno tudi razširiti, če jih želimo sploh uporabljati. Za delo s tovrstnimi podatki tako potrebujemo primerno programsko in strojno opremo. Stiskanje in razširjanje podatkov lahko namreč zahteva precejšnjo količino procesorske moči ter

pomnilnika (še posebej v primeru videa), glede na tip podatkov pa potrebujemo tudi temu primeren program.

Kakor se zvok in slika razlikujeta po svojih fizičnih lastnostih, se ta dva v digitalnem svetu razlikujeta po svoji vrsti zapisa. Vsega ne moramo stiskati po isti metodi, zato imamo na področju stiskanja podatkov na voljo več načinov in metod stiskanja. Glede na to »kaj« stiskamo, moramo vedeti tudi »kako« to stisniti. Na grobo ločimo dva načina stiskanja: izgubno in brez izgubno stiskanje.

4.1 Brez izgubno stiskanje

Z brez izgubnim stiskanjem podatke večinoma pretvarjamo v bolj učinkovit zapis, katerega pa lahko pozneje natančno pretvorimo nazaj v svojo prvotno obliko. Med postopkom stiskanja, brez izgubni algoritem naredi kodirno shemo, katera se nato uporabi za razširitev in rekonstrukcijo vsebine, v popolnoma enako obliko, kot je bila prvotno. Zaradi te natančne reprodukcije podatkov se tovrstno stiskanje večinoma uporablja za stiskanje teksta, kjer lahko izguba podatkov v tekstovnih datotekah privede do nedopustljivih semantičnih napak.

4.2 Izgubno stiskanje

Pri izgubnem stiskanju odstranjujemo podatke s katerimi posledično deformiramo celoten zapis. Razširjeni podatki s tem niso popolnoma identični prvotnemu zapisu podatkov, vendar smo s tem dosegli zelo dobro razmerje stiskanja, kar pomeni, da je lahko stisnjena datoteka bistveno manjša od prvotne. Ta način stiskanja se večinoma uporablja za stiskanje slik, zvoka in še posebej videa, zaradi obsežnih podatkov, ki si potrebni za njihovo predstavitev. Pri teh je namreč mogoče odstraniti podatke, ki jih človeška čutila ne zaznavajo, s čimer lahko ob primernem postopku stiskanja dosežemo to, da razlike med prvotno in stisnjeno vsebino, sploh ne opazimo. Tovrsten način stiskanja s tem večinoma bazira na razumevanju naših akustičnih in vizualnih sposobnosti. Neprimerna uporaba tovrstnega stiskanja pa lahko privede do zelo nekvalitetno stisnjenih vsebin, ki izgledajo in zvenijo popačeno. Pri uporabi izgubnega načina stiskanja, se rešitev načeloma skriva nekje na sredini, med razmerjem kakovosti zapisa in velikostjo datoteke, saj sta tako ena, kot druga skrajnost stiskanja po navadi neprimerni.

4.3 Metode stiskanja

Znotraj izgubnega in brez izgubnega načina, poznamo več metod stiskanja podatkov. Metode stiskanja predstavljajo način, s katerim pristopamo k stiskanju podatkov. Večina metod se deli v 4 kategorije: RLE kodiranje (angl. run length encoding - RLE), statistična (angl. statistical), z uporabo slovarja (angl. dictionary-based) in pretvorbena (angl. transforms). Izmed teh se vsaka metoda lahko namensko uporablja za svoj specifičen tip podatkov, saj se vsaka loti obdelave podatkov na svoj način. Te metode se lahko prav tako kombinirajo med seboj ali pa jih dopolnjujejo raznorazni kompresijski algoritmi. Obstajajo tudi druge metode, ki zaradi svojih posebnosti ne spadajo v nobeno od naštetih metod. (Salomon, Motta, 2010, 9)

Spoznajmo nekaj zelo pogosto uporabljenih metod oz. tehnik, za stiskanje podatkov:

Run-length Encoding (RLE)

RLE je enostavna metoda, ki združi vse zaporedne ponovitve znakovnih nizov v en števniki (število ponovitev) in en simbol (ponovljeni znak). Na primer: AAAAAA = 6A. RLE metoda se lahko uporablja za stiskanje vseh vrst podatkov (tekst, zvok, slika) in je strojno nezahtevna ter enostavna za implementacijo. (http://en.wikipedia.org/wiki/Run-length_encoding, dne 9.4.2011)

Aritmetično kodiranje

Aritmetično kodiranje (angl. arithmetic coding) je statistična metoda, ki celoten vhodni znakovni niz zakodira simbol za simbolom, v eno kodo, namesto tipičnega dodeljevanja posameznih kod simbolom. Je metoda, ki deluje počasneje, ampak zaradi dodatnega varčevanja s kodami, ki so potrebne za kodiranje, učinkovito. (http://en.wikipedia.org/wiki/Arithmetic_coding, dne 9.4.2011)

Predvidevanje z delnim ujemanjem

Predvidevanje z delnim ujemanjem (angl. prediction by partial matching, krajše PPM), je statistična metoda za stiskanje, ki izdelava seznam simbolov iz še ne stisnjenega zapisa podatkov, katerega pa pozneje uporabi za lažje predvidevanje simbolov v stisnjenem zapisu podatkov. (http://en.wikipedia.org/wiki/Prediction_by_partial_matching, dne 9.4.2011)

Kvantizacija (angl. quantization)

Je metoda, ki se večinoma uporablja za stiskanje slike in zvoka tako, da izvede preslikavo večje količine vhodnih podatkov, v manjši nabor podatkov. To stori tako, da med procesom stiskanja, vrednosti vhodnih podatkov posploši (zaokroži) do neke mere, s čimer se večje vrednosti pretvorijo v manjše. Poznamo skalarno in vektorsko kvantizacijo. Kvantizacija se lahko uporablja tudi za pretvarjanje analognega signala v digitalnega. (http://en.wikipedia.org/wiki/Quantization_%28signal_processing%29, dne 9.4.2011)

Ne glede na to kaj stiskamo in katero metodo smo pri tem uporabili, rezultat bo vedno drugačen zapis podatkov, kakor je bil pred tem. Pri stiskanju podatkov je tako nujno, da se z uporabo novo nastalega zapisa, uporabi ista metoda ali kodirna shema (algoritem), kakor je bila uporabljena za kodiranje tega, sicer novega zapisa podatkov ne moremo uporabiti.

4.4 Kompresijski algoritmi

Algoritem je postopek, ki z vrsto operacij vodi do rešitve določenega problema. V primeru stiskanja podatkov, nam ta narekuje, kako se podatki med stiskanjem in razširitvijo obdelujejo. Ta na nek način predstavlja načrt, kako se bodo podatki stisnili (kompresirali) in kako razširili (dekompresirali). Pri stiskanju podatkov nam algoritem predstavlja srce kompresijskega programa. Tovrstne algoritme lahko podjetja in organizacije razvijajo sami, lahko pa uporabijo že obstoječe algoritme, ki so bili razviti skozi čas in se zaradi njihove zanesljivosti in dobre učinkovitosti, uporabljajo še danes. Naše črke, znaki in jeziki na sploh, se skozi leta niso bistveno spremenili, zato se tudi algoritmi za stiskanje teksta niso. Podobno načelo drži za algoritme, ki so namenjeni stiskanju slike, zvoka in videa.

Spoznavmo nekaj najpogostejše uporabljenih kodirnih algoritmov:

Huffman

Je popularen brez izgubni kompresijski algoritem / metoda, katerega je leta 1952, razvil David A. Huffman. Algoritem deluje na principu dodeljevanja krajših kod znakom, ki se pojavljajo pogostejše in dodeljevanju daljše kode tistim znakom, ki se pojavljajo redkeje. Ob tem se izdela tako imenovano kodno drevo, ki služi za učinkovito shranjevanje kod in kasneje tudi za hiter razširitveni proces. Podoben algoritem predstavlja Shannon-Fano kodiranje. (http://en.wikipedia.org/wiki/Huffman_coding, dne 10.4.2011)

LZ77 in LZ78

Gre za univerzalna brez izgubna algoritma, ki sta bila leta 1977 in 1978, razvita od Abrahama Lempela in Jacoba Ziva. Algoritma generirata in uporabita slovar, s pomočjo katerega se nato nadomesti vse iste ponavljajoče gručne znakov, z enim krajšim znakom (ali znakovnim nizom) iz generiranega slovarja. Sta osnova za vse ostale variante LZ algoritmov, kot so LZW, LZSS, LZFG, LZMA,... ki so sledili. (http://en.wikipedia.org/wiki/LZ77_and_LZ78, dne 10.4.2011)

DEFLATE

Deflate je popularen algoritem, katerega je v osemdesetih razvil Phil Katz, in je uspešna kombinacija LZ77 in Huffmanovega algoritma skupaj. Temelji na uporabi drsnega slovarja. (<http://en.wikipedia.org/wiki/DEFLATE>, dne 10.4.2011)

Univerzalen in efektiven algoritem za stiskanje podatkov, ne obstaja. (Salomon, Motta, 2010, 8)

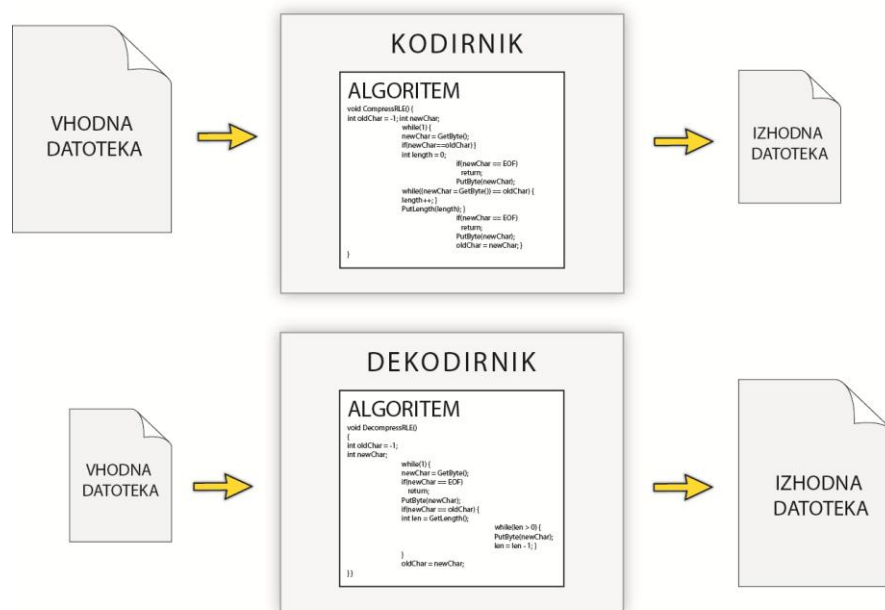
Kompresijske algoritme lahko delimo v dve kategoriji: simetrične in asimetrične. Ti se razlikujejo po tem, koliko dela opravi algoritem v procesu stiskanja in koliko za razširitev. **Simetrični algoritem** opravi približno enako količino dela pri stiskanju podatkov, kot ga opravi pri njihovi razširitvi, kar pomeni, da porabi podobno količino časa in procesorske moči za obe opravili. Takšen način je primeren, kadar je potrebno podatke stiskati ter razširjati istočasno, kot na primer pri spletnih video klicih. **Asimetrični algoritem** pa je zasnovan tako, da zahteva več dela pri enem procesu, kakor pri drugem. Stiskanje podatkov je lahko tukaj zahteven in dolgotrajen proces, medtem, ko je njihova razširitev hitra in enostavna. To je še posebej smiselno v primeru spletne vsebine, kjer se vsebina ob nastanku dobro stisne in je kasneje na voljo vsem le za ogled. Asimetrični algoritmi, ki delujejo v obratni smeri so redkejši, vendar prav tako uporabni. Izdelovanje rezervnih kopij na primer, je proces, ki mora potekati hitro in pogosto, uporaba rezervne kopije pa se zgodi le v redkih primerih.

4.5 Programi za stiskanje datotek

Pri stiskanju podatkov je algoritem neke vrste kodirna shema znotraj kodirnika in dekodirnika, po katerem se podatkovni tok stiska ali razširja. **Kodirnik** (angl. coder) je program, ki po svojem algoritmu, stisne voden tok podatkov in izdela stisnjen (manj redundanten) tok podatkov, kot rezultat. **Dekodirnik** (angl. decoder) pa je program, ki deluje

isto kot kodirnik, vendar v obratni smeri – razširi voden tok podatkov v razširjen (redundanten) tok podatkov. Za povprečnega uporabnika, ki podatkov ne stiska, ampak jih samo uporablja, je dovolj samo uporaba dekodirnika. (Salomon, Motta, 2010, 9)

Tukaj se pogosto pojavlja izraz **kodek** (angl. codec, izpeljano iz **coder/de**coder), ki predstavlja kodirnik in dekodirnik hkrati. Torej, če želimo preko interneta poslati nekomu stisnjene podatke, mora prejemnik za uporabo podatkov, na svojem sistemu imeti naložen primeren kodek oz. dekodirnik.



Slika 5: Princip delovanja kodirnika in dekodirnika (kodeka)

Vir: <http://www.cs.tut.fi/~albert/Dev/pucrunch/packing.html>, dne 2.4.2011; lasten

Kadar urejamo ali kreiramo naša besedila, slike, zvok ali video, nam tovrstni napredni programi, ker imajo v sebi že vgrajene kodirnike, možnosti stiskanja načeloma ponudijo med shranjevanjem podatkov v datoteko.

Datoteke lahko prav tako naknadno stiskamo s temu namenskimi programi. Raznorazni programi za stiskanje datotek, nam lahko poleg stiskanja podatkov nudijo tudi enkripcijo podatkov, pakiranje večjega števila datotek v en sam arhiv, pretvarjanje med datotečnimi formati in še več. Zaradi velikega števila formatov, so tovrstni programi prirejeni samo za stiskanje formatov, ki jih podpirajo. Tukaj programi za stiskanje teksta, načeloma uporabljajo razne brez izgubne algoritme, programi za stiskanje slik, zvoka in videa pa z različnimi

pretvorbami, odstranjujejo redundantne podatke. Kadar se srečamo s takšnimi programi, bomo v njih (če je program napisan v angleškem jeziku) zasledili angleške ukaze (izraze), kot so *encode* ali *compress* za stiskanje, ukaze *decode* ali *decompress* za njihovo razširitev, in *convert* ali *transcode*, za medsebojno pretvarjanje med različnimi vrstami formatov. Za podatke, ki še niso bili stisnjeni uporabljamo izraze, kot so nekompresirani, prvotni ali surovi (angl. raw) podatki, končni rezultat pa so stisnjeni, kompresirani ali kodirani podatki.

5 UČINKOVITOST STISKANJA PODATKOV

Na učinkovitost stiskanja podatkov vpliva več faktorjev. V igri je datotečni format, obsežnost in vsebina datoteke, vrsta zapisa, strojna oprema, ... najpomembnejšo vlogo pa igra algoritem, po katerem stiskamo. Ta se mora čim bolje soočiti z vsebino, ki jo mora stisniti. To je seveda tudi prvotni namen vseh tehnologij stiskanja podatkov, razvijati nove in izboljšati stare tovrstne algoritme. Učinkovitost algoritma lahko vrednotimo na več načinov. Lahko se osredotočimo na njegovo kompleksnost, strojno zahtevnost, hitrost obdelave podatkov, kompresijsko razmerje ali kako natančno rekonstruira prvoten zapis. Danes gre tukaj predvsem za hitrost ter kvaliteto stiskanja, in dober algoritem pomeni dober kompromis med tema dvema.

Pri razvoju kompresijskih algoritmov se razvijalci neprestano srečujejo z dilemo »kdaj je dovolj«. Neprestane izboljšave tovrstnih algoritmov lahko povzročijo zelo neprijetne posledice, saj se s kompleksnostjo algoritma večja tudi njegova zahtevnost. To pomeni, da lahko izboljšan algoritem, ki izboljša stiskanje za 1 %, poveča čas stiskanja in strojno zahtevnost za 10 %. Zaradi tega razloga se še danes vedno uporabljajo dobri stari kompresijski algoritmi, kot so Huffman ali Lemper-Ziv variante. (Salomon, Motta, 2010, 6)

Prav tako je nesmiselno stiskati majhne količine podatkov. Brez izgubno stiskanje dokumenta, ki vsebuje npr. samo eno poved, lahko privede do tega, da bo stisnjena datoteka celo večja od prvotne. V tem primeru lahko govorimo o negativni kompresiji, vzrok za to pa je kodirna shema, ki jo med stiskanjem izdelava kodirnik in prav tako zahteva nekaj prostora za njen obstoj. Stiskanje v izgubnem načinu pa lahko povzroči preveč popačenja.

Kaj pa če stisnjene podatke stisnemo večkrat? S takšnim početjem bi v primeru brez izgubnih načinov stiskanja, dosegli nič, saj so redundantni podatki že odstranjeni in nimamo ničesar več za stisniti oziroma odstraniti. S ponovnim stiskanjem podatkov, ki so stisnjeni po izgubnem načinu stiskanja pa lahko izrazito negativno vplivamo na kakovost zapisa, zato se to počne le takrat, kadar je stisnjena vsebina dovolj visoke kakovosti, da lahko s ponovnim stiskanjem žrtvujemo še nekaj kvalitete zapisa.

Pri analiziranju brez izgubnih načinov stiskanja tako govorimo o *natančnosti* in *kvaliteti* zapisov, medtem, ko se pri izgubnem načinu osredotočamo na tako imenovano *distorzijo* oziroma *popačenje*. (Sayood, 2000, 6)

5.1 Merjenje učinkovitosti

Eden izmed načinov merjenja učinkovitosti stiskanja je izračun **kompresijskega razmerja** po spodaj navedeni enačbi:

$$\text{kompresijsko razmerje} = \frac{\text{velikost izhodnega toka podatkov ali datoteke}}{\text{velikost vhodnega toka podatkov ali datoteke}}$$

Pri čemer lahko rezultat pomnožimo s 100 in dobimo količino stisnjenih podatkov v odstotkih, ki predstavljajo dejansko velikost prvotne datoteke ali toka. Kadar izračun preseže vrednost 1 ali preseže 100%, govorimo o negativni kompresiji.

Kompresijsko razmerje lahko dopolnimo z izračunom **prihranka** prostora, katerega je smiselno podati v odstotkih po sledeči formuli:

$$(1 - \text{kompresijsko razmerje}) \times 100 [\%]$$

ali

$$\text{prihranek prostora} = 1 - \frac{\text{velikost izhodnega toka podatkov ali datoteke}}{\text{velikost vhodnega toka podatkov ali datoteke}} \times 100 [\%]$$

Skozi nalogo bom svoje meritve podajal z izračunom kompresijskega razmerja in prihranka prostora, saj se mi zdita najbolj smiselna in pregledna.

Nasprotna meritev kompresijskega razmerja je **kompresijski faktor**, ki se mnogim zdi bolj naraven za tovrstne meritve, pri čemer vrednosti večje od 1 pomenijo večji kompresijski faktor in s tem tudi bolj učinkovito stisnjenost podatkov.

$$\text{kompresijski faktor} = \frac{\text{velikost vhodnega toka podatkov}}{\text{velikost izhodnega toka podatkov}}$$

(Salomon, Motta, 2010, 12)

Velikokrat zasledimo, da je kompresijsko razmerje podano tudi v obliki 200:1, 50:1, 2.6:1 in podobno. Tukaj gre za bolj posplošeno in včasih tudi dvoumno podajanje razmerja, kjer razmerje 2:1 pomeni, da je stisnjena datoteka velika polovico izvirne datoteke, medtem, ko kompresijsko razmerje 10:1, predstavlja desetino velikosti originalne datoteke ali toka podatkov.

6 TEKST V SPLETU IN NJEGOVE LASTNOSTI

Tekst je v spletu najverjetneje najbolj razširjen medij za podajanje informacij in komuniciranje. Spletne strani, dokumenti, elektronske knjige in podobni, vsi vsebujejo ogromne količine teksta. Ta se shranjuje v tekstovne datoteke, ob katerih sprva najverjetneje pomislimo na vse dokumente, ki so shranjeni na naših računalnikih, vendar ni tako. Tekstovne datoteke so vse računalniške datoteke, ki vsebujejo zaporedje linij. Struktura teh linij nato omogoča vnos znakov, ki so definirani po uporabljenem naboru znakov.



Slika 6: Struktura binarne tekstovne datoteke

Vir: Lasten

Stiskanje teksta zahteva drugačen pristop, kakor slika in zvok. Pri sliki in zvoku imamo večinoma opravka z izredno veliko količino binarnih vrednosti, ki predstavljajo analogne vrednosti. V primeru teksta pa so te vrednosti fiksno definirane. Ta je sestavljen iz črk, števil in simbolov, katere lahko z uporabo posebnih algoritmov in metod, pretvorimo v drugačno (manj redundantno) obliko zapisa. Stiskanje teksta zahteva odlično poznavanje informacijskih teorij. Te so se na tem področju začele razvijati že v poznih štiridesetih, za njihov povod pa je prispeval Claude Elwood Shannon. Strokovnjaki na tem področju so vedeli, da se nekatere besede in celo črke, skozi besedila pojavljajo pogosteje kakor druge, zato lahko te označimo in pretvorimo v krajši zapis. Tega sta se že okoli daljnega leta 1840, zavedala Samuel Morse in Alfred Vail, ki sta na tem principu izdelala znamenito Morsovo kodo. Ta koda prav tako uporablja krajše znake za črke, ki se pojavljajo pogosteje in daljše za tiste, ki se pojavljajo redkeje. (Sayood, 2000, 2)

Za stiskanje teksta se načeloma uporabljajo brez izgubni načini stiskanja in le v redkih primerih izgubni. Zaradi tega, bom tudi v tej nalogi predstavil samo brez izgubne metode.

Najpogostejše uporabljene metode za stiskanje teksta so statistične ali na slovarju bazirane metode, poznane tudi kot LZ metode. Ti algoritmi delujejo predvsem na principu pogostosti in verjetnosti znakov, ki se pojavljajo. Poglejmo si nekaj primerov za lažjo predstavo.

Teoretični primer 1

Pregovor Niccole Ugo Foscolota se glasi:

"Eni ljudje delajo in ne mislijo, drugi pa mislijo in ne delajo."

Stavek je sestavljen iz 12 besed, vsebuje 54 črk, dva ločila in 11 presledkov. Od tega hitro opazimo, da se besede *delajo*, *in*, *ne*, *mislijo* ponovijo, zato jih lahko označimo, kot redundantne in jim določimo drugačen zapis. Iz teh besed lahko naredimo manjši, svojevrsten slovar, v katerem bomo ponavljajoče besede zamenjali s krajšimi znaki.

Generiran slovar:

1 = delajo
2 = in
3 = ne
4 = mislijo

S tem se uporabljeni pregovor pretvori v: *"Eni ljudje 1234, drugi pa 4231."*

Uporabljena metoda temelji na osnovi slovarja (angl. dictionary-based method), kjer med postopkom stiskanja, kodirnik izdela slovar ponovljenih besed. Tega nato priloži v datoteko, kjer se kasneje lahko uporabi za njeno razširitev. Ta slovar sicer prav tako zasede nekaj prostora, a vendar se tovrstna metoda pri stiskanju obsežnejšega gradiva, izkaže za dokaj učinkovito. Algoritem, ki deluje po tej metodi je LZ78 algoritem.

Teoretični primer 2

12400000000XXXXhhhs

V primeru zgornjega 19 mestnega znakovnega niza opazimo, da se znaki *0*, *X* in *h* ponavljajo. Z uporabo RLE (angl. run-length encoding) metode, lahko zapis zapišemo v sledečem načinu:

1124804X3h1s ali za lažjo predstavo: 1(124),8(0),4(X),3(h),1(s)

S to metodo smo tako pridobili na količini uporabljenih znakov in njihovi velikosti, ki jo zasedajo. Nov zapis sedaj v stisnjeni obliki zasede 10 bajtov namesto 19.

6.1 HTTP kompresija

Spletne strani so večinoma podane v popularnih formatih, kot je HTML in njegova novejša različica XHTML. To so neke vrste dokumenti, ki so prilagojeni za spletne brskalnike. V dokumentu formata HTML je podano, kako se nam bo preko spletnega brskalnika prikazovala vsebina, ki jo ta dokument vsebuje. Ta je zapisana v označevalnem jeziku, ki lahko poleg teksta vsebuje vse od tabel, okvirjev, pozicij slik, povezav do drugih strani in datotek ter različnih skript (pHp, JavaScript, ASP.NET) in oblikovnih slogov (CSS), ki naredijo stran bolj dinamično in funkcionalno za uporabo. Vse skupaj je preko programskih jezikov prav tako podano v obliki teksta.

HTTP kompresija (angl. HTTP compression), je eden izmed načinov stiskanja spletnih strani. Tukaj se za stiskanje uporabi zelo standardizirane algoritme za stiskanje HTML/XHTML, CSS in JavaScript tekstovne vsebine, za pohitritev prenosa strani in za varčevanje pasovne širine. Za delovanje tovrstnega načina stiskanja mora stiskanje podpirati tako spletni strežnik (npr. Apache Web Server, Microsoft Internet Information Server - IIS), kot spletni brskalnik (npr. Internet Explorer, Mozilla FireFox, Google Chrome) in še sam HTTP protokol. S tem je vsebina na strežniku stisnjena še preden je poslana, spletni brskalnik pa medtem javi strežniku ali jo je sposoben prebrati. V primeru, da brskalnik ne podpira stisnjene vsebine, mu strežnik vrne prvotno (nekompresiorano) obliko vsebine. Vsebina na strežniku se večinoma stiska po deflate algoritmu s pomočjo *gzip* programa, njihovo uporabo in standardizacijo pa upravlja Internet Assigned Numbers Authority (IANA) organizacija. (<http://www.websiteoptimization.com/speed/tweak/compress/>, dne 3.5.2011)

O učinkovitosti HTTP kompresije je težko govoriti, saj je močno povezana z obremenjenostjo strežnika in podporo brskalnikov ter protokolov. Nekateri testi kažejo, da je z njo mogoče doseči tudi do 75 % prihranek na prostoru. Na splošno rečeno je priporočljiva za stiskanje obsežnih spletnih strani, podatkovnih baz in spletnih dokumentov. Stiskanje spletnih strani pa je izključno stvar administratorja in vzdrževalcev spletnih strani, na kar povprečen uporabnik spletna ne more imeti vpliva. Veliko spletnih strani pa sploh ni stisnjenih, saj zaradi skromne vsebine, napredne izdelave ali kompatibilnostih vzrokov, to ni potrebno. (<http://www.websiteoptimization.com/speed/tweak/compress/>, dne 3.5.2011)

6.2 Stiskanje tekstovnih datotek

Kakor sem že prej omenil, ima tekstovna datoteka širši pomen. Domači uporabnik se najverjetneje pogosteje sreča z datotekami formatov DOC, PDF, ODF ali celo TXT, ki so kreirane z različnimi urejevalniki dokumentov, kot so na primer Microsoft Word, OpenOffice.org Writer in podobni. Nekateri novejši tovrstni formati celo podpirajo nekatere metode stiskanja, kar med shranjevanjem. To pomeni, da lahko s primernimi nastavitvami (izbiro primerne datotečnega formata) znotraj urejevalnika dokumentov, vsebino dokumenta shranimo v že stisnjeni obliki dokumenta. Stiskanje spletnih strani in podatkovnih baz pa se vrši med obratovanjem strežnika in s strani programske opreme, ki ga poganja. Programe, ki jih razvijajo programerji so v osnovi prav tako tekstovne datoteke, ki vsebujejo programsko kodo in je lahko stisnjena s pomočjo namestitvenih programov, katere pogosto srečamo, kadar nameščamo program na naš operacijski sistem. Skratka obilo možnosti s strani razvijalcev programskih oprem in administratorjev.

Tekstovne datoteke so tako lahko programi, dokumenti, podatkovne baze, skripte ali datoteke z raznoraznimi podatki, ki se kljub svoji tekstovni osnovi lahko med seboj razlikujejo. Tem razlikam pa vsekakor ne more biti kos en sam kodirni algoritem, zato so kot univerzalna rešitev, tukaj na voljo posebni kompresijski programi. Programi, kot so WinZip, WinRar, 7-Zip, Stuffit in podobni, se uporabljajo predvsem za tovrstne reči. Ti vsebujejo več prirejenih kodekov oziroma kompresijskih algoritmov, s katerimi lahko stiskajo različne formate datotek, stisnjene datoteke pa nato »zapakirajo« v posebne arhivsko-kompresijske formate, kot so ZIP, RAR, 7z ali TAR. Ti programi in njihovi formati, delujejo na vseh popularnih operacijskih sistemih (Windows, Linux, Mac OS X), vendar je njihovo stiskanje datotek omejeno na formate, ki so jih sposobni stisniti. Nepodprte formate ne znajo stisniti, za vse podprte datotečne formate pa uporabljajo več možnih kompresijskih algoritmov, s katerimi so sposobni učinkovito stiskati razne tekstovne in celo slikovne datoteke, prav tako pa so sposobni za malenkost stisniti nekatere zvočne in video formate.

6.3 Razširjeni arhivsko-kompresijski formati

6.3.1 ZIP

Zip je izredno popularen arhivsko-kompresijski format, katerega je leta 1989 razvil Phil Katz z uporabo svojega deflate algoritma. Za njegovo uporabo je sprva bilo potrebno imeti naložen

program PKZIP, kasneje pa so uporabo formata implementirali tudi v vse popularne operacijske sisteme. Datotečni format je sposoben stiskati več datotek in različnih formatov hkrati ter jih skupaj združiti v eno ZIP datoteko, pri čemer je še posebej učinkovit s tekstovnimi datotekami. (http://en.wikipedia.org/wiki/ZIP_%28file_format%29, dne 15.5.2011)

- Polno ime: ZIP (pomeni hitrost)
- Končnica datoteke: .zip
- Lastnosti: brez izgub, odprt format

6.3.2 RAR

Arhivsko-kompresijski format, ki je priljubljen predvsem po Evropi, Rusiji ter Aziji, je lastniški format podjetja Rarlabs, katerega je leta 1993 po svojevrstnem algoritmu razvil Ruski programer Eugene Roshal. Format, v katerega je mogoče shranjevati le z uporabo programa WinRAR, je sposoben boljše kompresije od formata ZIP, vpenjanja datotek in korekcije napak. (<http://sl.wikipedia.org/wiki/RAR>, dne 15.5.2011)

- Polno ime: Roshal Archive (**R**oshal **A**Rchive)
- Končnica datoteke: .rar, .rev, .r## (.r00, .r01, .r02,r99)
- Lastnosti: brez izgub, zaprt format

6.3.3 7z

Odprt arhivsko-kompresijski format, katerega je 1999 na trg pospremil prav tako odprt in brezplačen program 7-Zip, je delo Ruskega programerja Igorja Pavlova. Format, ki je z serijo modificiranih algoritmov (LZMA algoritem) sposoben celo boljšega kompresijskega razmerja kakor konkurenca, kot pravijo, je prav tako sposoben enkripcije ter ostalih naprednih funkcij. Na žalost pa trpi z nepodprtostjo s strani operacijskih sistemov in neprepoznavnostjo med uporabniki. Kljub temu ta predstavlja odlično brezplačno alternativo programu WinZip in WinRAR. (<http://en.wikipedia.org/wiki/7z>, dne 15.5.2011)

- Polno ime: 7z
- Končnica datoteke: .7z
- Lastnosti: brez izgub, odprt format

6.4 Učinkovitost stiskanja tekstovnih datotek

Za stiskanje bom uporabil eno tipično tekstovno datoteko (On the Origin of Species.txt), ki vsebuje večjo količino besedila (celotno knjigo) in jo pretvoril v dva najbolj razširjena tekstovna formata, ki krožita po spletu: DOC in PDF. Datoteke, ki bom stiskal so sledeče:

1. On the Origin of Species.txt

Vir: <http://www.free-ebooks.net/ebook/On-the-Origin-of-Species/txt>, dne 17.5.2011 | format: Plain Text

2. On the Origin of Species.doc

Vir: <http://www.free-ebooks.net/ebook/On-the-Origin-of-Species/txt>, dne 17.5.2011 | format: Microsoft Word Document

3. On the Origin of Species.pdf

Vir: <http://www.free-ebooks.net/ebook/On-the-Origin-of-Species/txt>, dne 17.5.2011 | format: Portable Document Format

Sledil pa bo preizkus stiskanja malce drugačnih tekstovnih formatov in stiskanje mape z različno tekstovno vsebino. Med njimi bom stiskal tudi datoteko z formatom DOCX, ki je naslednik formata DOC.

4. On the Origin of Species.docx

Vir: <http://www.free-ebooks.net/ebook/On-the-Origin-of-Species/txt>, dne 17.5.2011 | format: Microsoft Word Document

5. Data_compression.htm

Vir: http://en.wikipedia.org/wiki/Data_compression, dne 17.5.2011 | format: Hyper Text Markup

6. Plant_catalog.xml

Vir: http://www.w3schools.com/xml/plant_catalog.xml, dne 17.5.2011 | format: Extensible Markup Language

7. YUI! Library 2 (Full Developer Kit)

Vir: <http://developer.yahoo.com/yui/2/>, dne 17.5.2011 (mapa z datotekami)

8. VLC Portable

Vir: http://portableapps.com/apps/music_video/vlc_portable, dne 17.5.2011 (mapa z datotekami)

Uporabljeni programi

Za preizkus učinkovitosti pri stiskanju tekstovnih datotek, bom uporabil tri programe: vsesplošno znan WinZip, konkurenčni WinRAR in brezplačni 7-Zip. Našteti programi

podpirajo več formatov za stiskanje, omogočajo več nivojev stiskanja in med tem uporabljajo več različnih kompresijskih algoritmov. Za pretvarjanje med datotečnimi formati dokumentov sem uporabil *Microsoft Office Word 2010*. Kot alternativa lahko tukaj uporabimo *OpenOffice.org Writer*.

Praktični preizkus 1

DATOTEKA: On the Origin of Species.txt, velikost: 1,12 MB (1.179.299 B)				
PROGRAM	KOMPRESIJSKI FORMAT	NIVO STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
WinZip	ZIP (.zip)	Standardni	350 KB (359.316 B)	0,30 [70 %]
		Najvišji	366 KB (375.623 B)	0,32 [68 %]
WinRAR	RAR (.rar)	Standardni	306 KB (313.813 B)	0,27 [73 %]
		Najvišji	225 KB (230.718 B)	0,20 [80 %]
7-Zip	7z (.7z)	Standardni	284 KB (290.839 B)	0,25 [75 %]
		Najvišji	283 KB (290.468 B)	0,25 [75 %]
DATOTEKA: On the Origin of Species.doc, velikost: 1,20 MB (1.268.736 B)				
WinZip	ZIP (.zip)	Standardni	375 KB (384.688 B)	0,30 [70 %]
		Najvišji	393 KB (402.504 B)	0,32 [68 %]
WinRAR	RAR (.rar)	Standardni	321 KB (328.821 B)	0,26 [74 %]
		Najvišji	219 KB (244.829 B)	0,19 [81 %]
7-Zip	7z (.7z)	Standardni	298 KB (305.724 B)	0,24 [76 %]
		Najvišji	299 KB (306.027 B)	0,24 [76 %]
DATOTEKA: On the Origin of Species.pdf, velikost: 1,17 MB (1.228.428 B)				
WinZip	ZIP (.zip)	Standardni	1,05 MB (1.107.181 B)	0,90 [10 %]
		Najvišji	1,05 MB (1.107.328 B)	0,90 [10 %]
WinRAR	RAR (.rar)	Standardni	1,07 MB (1.105.627 B)	0,90 [10 %]
		Najvišji	1,05 MB (1.105.127 B)	0,90 [10 %]
7-Zip	7z (.7z)	Standardni	1,06 MB (1.110.151 B)	0,90 [10 %]
		Najvišji	1,06 MB (1.109.271 B)	0,90 [10 %]

Tabela 1: Preizkus stiskanja TXT, DOC in PDF tekstovnih formatov

Vir: Lasten

Praktični preizkus 2

Sledi preizkus stiskanja različnih tekstovnih datotek z najuspešnejšim programom in njegovim uporabljenim formatom, iz prejšnjega preizkusa. Uporabljen program je WinRAR.

DATOTEKA / MAPA	VELIKOST	NIVO STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
On the Origin of Species.docx	484 KB (496.205 B)	Standardni	483 KB (494.773 B)	0,9 [0,1 %]
		Najvišji	483 KB (494.777 B)	0,9 [0,1 %]
Data_compression.htm	252 KB (259.014 B)	Standardni	46 KB (48.028 B)	0,19 [81 %]
		Najvišji	46 KB (47.295 B)	0,18 [82 %]
Plant_catalog.xml	7,37 KB (7.547 B)	Standardni	1,2 KB (1.228 B)	0,16 [84 %]
		Najvišji	1 KB (1.040 B)	0,14 [86 %]
YUI! Library*	78,0 MB (81.846.680 B)	Standardni	10,7 MB (11.324.872 B)	0,14 [86 %]
		Najvišji	9,20 MB (9.648.923 B)	0,12 [88 %]
VLC Portable*	35,8 MB (37.618.149 B)	Standardni	22,6 MB (23.785.821 B)	0,63 [37 %]
		Najvišji	22,6 MB (23.729.331 B)	0,63 [37 %]

* mapa, ki vsebuje več tipov tekstovnih datotek, kot so programi, skripte, dokumenti ipd.

Tabela 2: Stiskanje različne tekstovne vsebine s programom WinRAR

Vir: Lasten

6.5 Ugotovitve

Iz preizkusa je razvidno, da se učinkovitost stiskanja lahko razlikuje od uporabljenih programov in formatov, vendar odstopanja niso pretirano velika. Višji nivo stiskanja se lahko v nekaterih primerih izkaže celo slabše, kot standardni nivo, bistvene razlike med njima pa ni. Zelo dobro se je izkazal program WinRAR in njegov RAR format. Ta je za razliko od konkurence malce počasnejši med stiskanjem, vendar se ta dodatno vložen čas, na koncu tudi obrestuje. Iz preizkusa je prav tako razvidno, kako se različni formati datotek stiskajo. Enostavne tekstovne datoteke, kot so TXT, HTML ali XML, je mogoče zaradi svoje enostavne strukture izredno dobro stisniti. Pri nekaterih lahko z njihovim stiskanjem privarčujemo tudi do 80 % ali več prostora. Nekateri formati, kot so PDF ali DOCX pa se stiskajo bistveno slabše. Tukaj je bilo PDF zaradi svoje strukture mogoče stisniti le za 10 % svoje prvotne velikosti, DOCX format pa za slabih 0,1 %. Vzrok za tako slabo stiskanje tiči v strukturi teh datotečnih formatov, saj že sami podpirajo stiskanje podatkov. Tukaj je PDF že ob shranjevanju optimiziran za varčevanje s prostorom, DOCX pa je v bistvu prirejen ZIP format, ki vsebuje XML elemente za shranjevanje vsebine. (<http://www.fileinfo.com/extension/docx>, dne 20.5.2011)

Predlogi

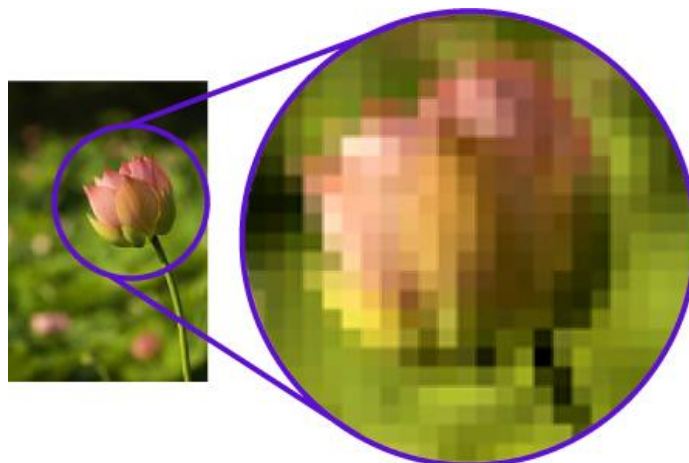
Programe, ki sem tukaj uporabil za stiskanje bom opredelil sledeče: WinZip za popolno podporo, WinRAR za visoko učinkovitost in 7-Zip za odprtost ter brezplačnost. Višji nivoji stiskanja ne dajo bistveno boljših rezultatov, znajo pa podaljšati čas stiskanja, zato jih odsvetujem. Za kompleksnejše formate kot so PDF ali DOCX pa menim, da jih je bolje pustiti v svoji prvotni obliki, saj z njihovim stiskanjem ne bomo privarčevali veliko. Mogoče s stiskanjem večjega števila PDF dokumentov, kjer bi se 10 % prihranka na dokument že poznalo. Priporočam stiskanje in pakiranje večjega števila datotek, ki so lahko celo mešanih formatov. Ti sposobni programi znajo namreč brez izgubno stiskati tudi različne slikovne, zvočne in video datoteke.

7 SLIKE V SPLETU IN NJIHOVE LASTNOSTI

Danes so slike tiste, ki dajejo spletnim stranem videz in dopolnjujejo njihovo vsebino. Te se na spletu pojavljajo kot ozadja, meniji, reklame, gumbi, ločilne črte in fotografije, pri čemer jih je zaradi njihove pogostosti uporabe, potrebno nujno stisniti oziroma optimizirati. Slike, ki se uporabljajo v spletu so pripravljene drugače, kakor slike, ki so namenjene tiskanju. Tukaj je potrebno upoštevati lastnosti računalniških zaslonov, spletnih brskalnikov in strukturo same spletne strani. Za stiskanje slik prav tako obstajajo različne metode stiskanja v izgubnem in brez izgubnem načinu. Pomembno vlogo predstavlja tudi datotečni format slike. Preden pa spoznamo načine za stiskanje slik, si oglejmo še dve podatkovne strukture, s katerimi so slike v računalništvu predstavljene.

7.1 *Rastrska slika*

Rastrske ali bitne slike (angl. bitmap) imajo podatkovno strukturo, ki je predstavljena kot mreža barvnih pik (angl. pixels). Takšno sliko sestavlja x vrstic in y stolpcev barvnih pik, ki kot celota tvorijo podobo slike. Z množenjem teh pik ($x \times y$) dobimo ločljivost slike, ki z njeno širino in višino, predstavlja njeno velikost. Večje število pik pomeni večjo ločljivost slike, večja ločljivost pa večjo kvaliteto in posledično tudi večjo datoteko za predstavitev slike. Barvne pike so lahko predstavljene z različnimi barvnimi modeli, kot na primer RGB ali CMYK, ki na različne načine predstavljajo barve v sliki. Glede na barvno globino slike lahko določimo število barv, ki jih lahko prikaže ena barvna pika. 8 bitna slika na primer, je tako sposobna prikazati $2^8 = 256$ število barv, medtem, ko jih je 24 bitna slika sposobna prikazati $2^{24} = 16.777.216$ število barv. To je 256 odtenkov rdeče, zelene in modre barve po RGB barvnem modelu, kar zadostuje tudi sposobnostim človeškega očesa. (Salomon, Motta, 2010, 447)



Slika 7: Rastrska slika s primerom izredne povečave

Vir: http://vector-conversions.com/images/raster_vs_vector_1.jpg, dne 2.8.2011

Na splošno in še posebej na spletu, prevladuje uporaba rastrskih slik, saj so primerne za naravno prikazovanje podrobnosti in barv. Tovrstna predstavitev slike pa ima svoje slabosti. Povečava rastrske slike preko njene originalne velikosti, pomeni izračun in prikaz dodatnih barvnih pik, kar posledično povzroči vidno poslabšanje kvalitete slike (izguba ostrine), izredna pomanjšava pa popačenja. Prekomerno izgubno stiskanje rastrskih slik, lahko povzroči izris tako imenovanih kompresijskih artefaktov, ki prav tako vplivajo na kvaliteto slike.

7.2 Vektorske slike

Vektorske slike za prikaz podobe uporabljajo matematično definirane geometrijske elemente kot so pike, črte, krivulje, geometrijske like in poligone. Ti so znotraj datoteke predstavljeni kot programska koda, kombinirana z matematičnimi izrazi, koordinatami poti in barvnimi kodami. Prednost vektorskih slik je ta, da s povečanjem ali manjšanjem slike, kvaliteta ostaja enaka, saj se matematični elementi v tem procesu preračunajo in ponovno izrišejo. S tem so posledično datoteke takšnih slik, tudi manjše, saj so v njih zapisane le matematične formule, za prikaz elementov v sliki. Zaradi tovrstne strukture izgledajo te slike manj naravne, vendar so izredno uporabne za izdelavo logotipov, risanje slik in 3D modeliranje.



Slika 8: Primer vektorsko narisane avtomobila

Vir: <http://www.edrawsoft.com/images/cliparts/clipart-car.png>, dne 3.8.2011

Zaradi prevlade rastrskih slik na spletu, bo v tej diplomski nalogi poudarek na njih. Kljub temu pa bom opravil manjši preizkus učinkovitost stiskanja vektorskih slik ter predstavil zelo razširjen SVG (Scalable Vector Graphic) format, za shranjevanje vektorskih slik. Uporaba vektorskih slik v spletu se dandanes počasi večja.

Vektorske slike se načeloma stiskajo po brez izgubnih načinih. Ker pa predstavljajo neke vrste tekstovne datoteke, jih je možno stiskati tudi s kompresijskimi programi, kot so WinZip, WinRAR, 7-Zip in podobni.

7.3 Stiskanje in optimiziranje slik

Za slike, ki se obdelujejo za uporabo v spletu, se velikokrat reče, da sliko optimiziramo. To je zaradi tega, ker se poleg stiskanja slike, izbere tudi temu primeren format in velikost. Za tovrstno optimiziranje je potrebno uporabiti naprednejše programe za obdelavo in izdelavo slik. Večina izdelovalcev spletnih strani se tako slej kot prej sreča s programi, kot so na primer Adobe Photoshop ali Fireworks, GIMP, Corel Photo-Paint in podobni, s katerimi optimizirajo slike tako, da so primerne za uporabo v spletu. Slike v spletu so načeloma manjših velikosti (ločljivosti) in so shranjene v primernem datotečnem formatu glede na to ali gre za sliko za ozadje, fotografijo, animacijo ali enostavno grafiko. Prikaz spletnih slik je odvisen tudi od samega spletnega brskalnika in njegove podpore za prikaz grafičnih datotečnih formatov.

Predstavljajmo si rastrsko sliko širine 800 pik in višine 600 pik ter 24 bitne barvne globine, ki jo shranimo brez stiskanja.

$$800 \times 600 \times 24 = 1.440.000 \text{ bajtov (1.37 MB)}$$

Takšna slika se danes smatra za sliko manjše ločljivosti, a vendar bi ta pred leti skoraj popolnoma zapolnila eno 3.5" disketo. Njen prenos preko interneta bi preko tipičnega 56 Kbps modema, trajal dobri dve minuti. Danes s spleta prenesemo takšno sliko v nekaj sekundah, njena velikost na pomnilniški kartici ali USB ključu pa bi bila zanemarljiva. Zakaj torej stiskati slike? Odgovor je v razvoju informacijskih tehnologij in novimi navadami, ki smo jih ob tem razvili. Danes namreč napravimo bistveno več slik večjih ločljivosti, te pa nato v večjem številu pošiljamo preko spleta. Brez stiskanja teh slik bi takšno dejanje močno bremenilo promet, prenosi pa bi trajali neprijetno dolgo. Ne pozabimo tudi na to, da nam večina internetnih ponudnikov nudi višje hitrosti prenosa pri prenašanju, kot pa pri oddajanju datotek.

Poglejmo si torej osnovni princip stiskanja rastrske slike v brez izgubnem in izgubnem načinu. Kadarkoli izberemo poljubno barvno piko v sliki, je velika verjetnost, da bo sosednja pika podobne ali iste barve.

Teoretični primer 1, brez izgubno stiskanje

Spodnja črta predstavlja vrstico neke slike, ki je predstavljena iz zaporedja zelenih, črnih in modrih barvnih pik:



Slika 9: Iz različnih barvnih pik sestavljena črta

Vir: Lasten

Vrstico tako sestavlja 24 zelenih pik (Z), 11 črnih pik (Č) in 12 modrih pik (M). S preprosto RLE metodo stiskanja, lahko ta zapis pretvorimo v sledečo obliko:

$$24(Z), 11(\check{C}), 12(M)$$

Ali za lažjo predstavo, natančneje definiran zapis po RGB barvnem modelu:

$24(R34,G177,B76),11(R0,G0,B0),12(R0,G162,B232)$

Če uporabimo isto metodo stiskanja za vseh ostalih vrstic slike, bi kot rezultat dobili kar učinkovito stisnjeno sliko, pri čemer ne bi utrpeli nobene izgube podatkov.

Teoretični primer 2, izgubno stiskanje

S stiskanjem slike po izgubnem načinu večinoma izkoriščamo fizične lastnosti človeškega vida. Ta je bolj občutljiv na spremembe svetilnosti barve, kot pa na njen ton.



Slika 10: Izvirna slika na levo in slika stisnjena v JPEG formatu na desno

Vir: Lasten

S tovrstnim stiskanjem smo sicer izgubili nekaj barv, vendar je datoteka stisnjene slike s tem skoraj za polovico manjša. Za še bolj učinkovito stiskanje slike pa se v izgubnem načinu uporabljajo še dodatne metode in algoritmi. En dodaten način je delitev slike na več manjših matric, v katerih se barvne pike še dodatno posplošijo in na ta način tudi stisnejo. Takšno dejanje pa že privede do kompresijskih artefaktov in izrazitejših popačenj v sami sliki.



Slika 11: Posledice vedno bolj intenzivnega stiskanja iz leve proti desni

Vir: http://en.wikipedia.org/wiki/File:JPEG_example_JPG_RIP_001.jpg, dne 21.8.2011,

http://en.wikipedia.org/wiki/File:JPEG_example_JPG_RIP_010.jpg, dne 21.8.2011 in

http://en.wikipedia.org/wiki/File:JPEG_example_JPG_RIP_025.jpg, dne 21.8.2011

7.4 Razširjeni grafični formati

Metode za stiskanje tovrstnih slik so omejene na uporabljen slikovni datotečni format. Tukaj ima vsak svoje prednosti in slabosti, uporaba drugih (nestandardnih) formatov pa lahko privede do nepodprtosti s strani spletnega brskalnika. Slike, katere zasledimo na spletu, so tako večinoma JPEG, GIF ali PNG datotečnega formata. Imajo prilagojene ločljivosti, ki so primerne za prikaz na računalniškem zaslonu in so izdelane po RGB barvni shemi. Med temi se vektorske slike načeloma prihajajo v SVG formatu.

7.4.1 JPEG

JPEG ali krajše JPG, je najpogostejše uporabljan datotečni format za prikazovanje slik na spletu in še marsikje drugje. Prva različica je bila razvita leta 1992 od združenega odbora ISO in CCITT (danes poznani kot ITU-T), ki so se skupaj poimenovali Joint Photographic Experts Group - JPEG, kar predstavlja tudi polno ime tega datotečnega formata. Gre za izredno razširjen in fleksibilen izgubni format, ki je sposoben prikazovati slike z 24 bitno barvno globino. Podpira več različnih barvnih palet in je izredno primeren za prikazovanje barvnih fotografij, a neprimeren za prikaz enobarvnih. Podpira integracijo meta podatkov (na primer JPEG/exif pri večini fotoaparatorov) in je sposoben doseči visoko kompresijsko razmerje tako, da omogoča uporabniku nastavitve kompresijskih parametrov po svoji meri. To je tudi največja prednost tega formata. Za stiskanje podatkov ta uporablja metodo reduciranja barv s pretvarjanjem med različnimi barvnimi modeli in z grupiranjem barvnih pik v skupine, katere nato še dodatno stisne. Obstaja tudi možnost brez izgubnega stiskanja v tem formatu, vendar

se zaradi nizkega kompresijskega razmerja, uporablja redkeje. (<http://en.wikipedia.org/wiki/Jpeg>, dne 23.8.2011)

- Polno ime: Joint Photographic Experts Group
- Končnica datoteke: .jpg, .jpeg, .jpe, .jif, .jfif, .jfi
- Lastnosti: izguben z možnostjo brez izgubnega, odprt format

7.4.2 GIF

GIF ali GIFF format, je delo podjetja CompuServe Information Services, ki ga je svetu predstavilo leta 1987. Gre za format, ki podpira največ 8 bitno barvno globino, s čimer je sliko možno prikazati z največ 256 barvami po RGB barvnem modelu. Zaradi takšne barvne omejitve format ni primeren za prikazovanje kompleksnejših barvnih fotografij, vendar odigra dobro vlogo s prikazovanjem enostavnejših slikovnih elementov, kot so enobarvne grafike, logotipi ali znaki. Originalna različica tega formata je bila imenovana 87a, kasneje pa je ga je podjetje izpopolnilo z različico 89a, ki je sposobna podpirati sekvenco zaporednih slik (animacijo), enostavno obliko prosojnosti in še nekaj dodatkov. Ta temelji na brez izgubnem LZW (Lempel-Ziv-Welch) kompresijskem algoritmu, za katerega podjetje CompuServe ni vedelo, da si patent lasti podjetje Unisys. Ta je kasneje, ko je datotečni format GIF postal masovno uporabljan, leta 1994 sklenilo dogovor s CompuServe, ki je zahteval nakup licence, za uporabo tega datotečnega formata, v komercialne namene. To dejanje je sprožilo razvoj novega formata, katerega opis kmalu sledi. (Sayood, 2000, 134)

- Polno ime: Graphic Interchange Format / Graphic Interchange File Format
- Končnica datoteke: .gif
- Lastnosti: brez izguben, zaprt format

7.4.3 PNG

Datotečni slikovni format PNG, se je začel razvijati leta 1995, kot alternativa popularnemu GIF formatu, ki je takrat podlegel sporu patentiranega kompresijskega algoritma, ki ga je vseboval. Ta je tako nastal kot projektno delo skupine različnih strokovnjakov s tega področja in leta 1996 tudi zaživel. PNG podpira barvno globino do 24 bitov po RGB barvnem modelu, ki je lahko kombiniran z dodatnim 8 bitnim alfa kanalom (RGBA), za napreden prikaz prosojnosti. Kadar ju uporabimo skupaj, to pomeni, da gre za sliko 32 bitne barvne globine. Za stiskanje podatkov uporablja kompresijski algoritem deflate. PNG sicer ne podpira

animacije, ampak so namesto tega zato kasneje razvili na njemu osnovan MNG (Multiple-image Network Graphic) format, ki je namenjen izključno animacijam. Danes je PNG izredno razširjen in priznan kot standard za prikazovanje spletnih slik. (Sayood, 2000, 135)

- Polno ime: Portable Network Graphic
- Končnica datoteke: .png
- Lastnosti: brez izguben, odprt format

7.4.4 SVG

SVG je format, ki temelji na XML datotečnem formatu. Ta se uporablja za prikazovanje dvodimenzionalnih (2D) statičnih in dinamičnih vektorskih slik. Je odprt datotečni format, ki ga razvija World Wide Web Consortium (W3C) že od leta 1999 dalje. Je zelo fleksibilen in sposoben format, kateri sprva ni bil deležen velike podpore s strani spletnih brskalnikov, danes pa ga popularni brskalniki kot so Mozilla FireFox, Internet Explorer 9, Google Chrome in Safari, že v zadovoljivi meri podpirajo. (<http://en.wikipedia.org/wiki/SVG>, dne 23.8.2011)

- Polno ime: Scalable Vector Graphics
- Končnica datoteke: .svg, .svgz
- Lastnosti: brez izguben, odprt format

7.5 Učinkovitost stiskanja slikovnih datotek

S sledečim preizkusom, bom najprej poskusil razkriti učinkovitost treh najbolj razširjenih formatov za rastrske slike na spletu: JPEG, GIF in PNG. Med stiskanjem bo izgubni format JPEG deležen še podrobnejšega preizkusa z različnimi nivoji stiskanja. Kasneje pa bom opravil še en krajši preizkus, s stiskanjem vektorskih slik. Za test bom uporabil eno rastrsko sliko in dve vektorski sliki:

1. Zadnji vhod.cr2

Vir: Canon EOS 50D | ločljivost: 4752 × 3168 | barvna globina: 48 bitna

2. VectorBitmapExample.svg

Vir: <http://en.wikipedia.org/wiki/File:VectorBitmapExample.svg>, dne 24.8.2011

3. Watermelon.svg

Vir: <http://www.openclipart.org/detail/131845/water-melon-by-gnokii>, dne 24.8.2011

Slika zajeta s fotoaparatom Canon EOS 50D, je bila shranjena v Canonovem surovem (RAW) CR2 formatu. Zaradi hitrejše obdelave, bom sliko pomanjšal na ločljivost 2376×1584 , ter jo pretvoril v tri barvne tipe: barvna (24 bitna barvna globina), sivinska (8 bitna barvna globina) in črno-bela (1 bitna barvna globina). Ob teh spremembah bo potrebno sliko pretvoriti v nekompresiran BMP (Bitmap Image File) format, saj je zaradi nepodprtosti CR2 formata, sicer to nemogoče.



Slika 12: Stiskanje barvne, sivinske in črno-bele slike

Vir: Lasten

Uporabljeni programi

Za pretvarjanje med formati si bom pomagal s programom *ACDSee Photo Manager 12*, za vse dodatne funkcije ali obdelavo slike, če bo potrebno pa bom uporabil *Adobe Photoshop*. Kot brezplačna alternativa bi tukaj priporočal program *FastStone Image Viewer* in *Gimp*. Za stiskanje vektorskih slik pa bom uporabil program *WinRAR*, za stiskanje datotek v arhivsko-kompresijskem RAR formatu ter program *Adobe Illustrator*, za shranjevanje slik v kompresijskem SVGZ formatu. SVGZ format je praktično SVG format, stisnjen po gzip kompresijskem algoritmu. Kot brezplačna alternativa programu *Illustrator*, priporočam program *Inkscape*.

Praktični preizkus 1

DATOTEKA: Zadnji vhod - barvna.bmp, velikost: 10,7 MB (11.290.806 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
GIF	/	1,90 MB (1.996.815 B)	0,18 [82 %]
PNG	/	6,15 MB (6.458.341 B)	0,57 [43 %]
JPEG	100	3,57 MB (3.744.129 B)	0,33 [67 %]
	70	646 KB (661.765 B)	0,06 [94 %]
	50	457 KB (468.892 B)	0,04 [96 %]

Tabela 3: Stiskanje barvne slike

Vir: Lasten

DATOTEKA: Zadnji vhod - sivinska.bmp (Bitmap Image), velikost: 3,58 MB (3.764.662 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
GIF	/	3,51 MB (3.686.100 B)	0,98 [2 %]
PNG	/	2,98 MB (3.125.579 B)	0,83 [17 %]
JPEG	100	2,08 MB (2.188.804 B)	0,58 [42 %]
	70	561 KB (575.041 B)	0,15 [85 %]
	50	404 KB (413.933 B)	0,11 [89 %]

Tabela 4: Stiskanje sivinske slike

Vir: Lasten

DATOTEKA: Zadnji vhod - crnobela.bmp (Bitmap Image), velikost: 1,79 MB (1.888.190 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
GIF	/	596 KB (610.316 B)	0,32 [68 %]
PNG	/	522 KB (535.403 B)	0,28 [72 %]
JPEG	100	5,72 MB (6.001.163 B)	3,18 [-218 %]
	70	3,27 MB (3.437.743 B)	1,82 [-82 %]
	50	2,78 MB (2.922.646 B)	1,55 [-55 %]

Tabela 5: Stiskanje črno-bele slike

Vir: Lasten

Praktični preizkus 2

S tem drugim preizkusom želim ugotoviti, kako učinkovito se stiskajo rastrske slike pri nižjih ločljivosti. Za preizkus bom sliko »Zadnji vhod – barvna.bmp«, pomanjšal na ločljivost 500 × 333.

DATOTEKA: Zadnji vhod – Bx333.bmp (Bitmap Image), velikost: 487 KB (499.554 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
GIF	/	99,4 KB (101.835 B)	0,20 [80 %]
PNG	/	311 KB (319.228 B)	0,64 [36 %]
JPEG	100	182 KB (187.090 B)	0,37 [63 %]
	70	42,7 KB (43.728 B)	0,09 [91 %]
	50	31,2 KB (32.005 B)	0,06 [94 %]

Tabela 6: Stiskanje barvne slike nizke ločljivosti

Vir: Lasten

Praktični preizkus 3

Sledeči preizkus je namenjen preverjanju učinkovitosti stiskanja slik v vektorskem zapisu, ki so shranjene v SVG formatu.

DATOTEKA	VELIKOST PRED STISKANJEM	UPOR. KOMPRES. FORMAT	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
VectorBitmapExample.svg	114 KB (117.502 B)	SVGZ	35,5 KB (36.410 B)	0,31 [69 %]
		RAR	32,8 KB (33.621 B)	0,29 [71 %]
Watermelon.svg	52,7 KB (54.048 B)	SVGZ	9,06 KB (9.286 B)	0,17 [83 %]
		RAR	52,7 KB (54.048 B)	0,22 [78 %]

Tabela 7: Stiskanje SVG (Scalable Vector Graphic) datotek

Vir: Lasten

7.6 Ugotovitve

Slike večjih ločljivosti je mogoče zelo dobro stisniti brez izrazitejših popačenj ali artefaktov, medtem, ko lahko slike manjših ločljivosti hitro povzročajo vidna popačenja. Glede na uporabljene barvne globine s katerimi so bile slike predstavljene, so se uporabljeni slikovni formati odzivali različno. Tukaj se je med stiskanjem sivinske slike najslabše izkazal GIF, pri stiskanju eno bitne črno-bele slike pa se je JPEG z negativnimi rezultati kompresiranja, izkazal kot popolni polom. S strani kvalitete stiskanja se je najbolje izkazal PNG format, ki pa žal ne nudi vedno visokega razmerja stiskanja. Z visokim razmerjem stiskanja zna presenetiti brez izgubni GIF format, ki pa zaradi svojih barvnih omejitev, povzroča izrazitejšo zrnatost v slikah.

Vektorske slike formata SVG je možno učinkovito stisniti z uporabo kompresijskega formata SVGZ, kot kompresijskimi programi WinRAR, WinZip, 7-Zip ter podobnimi.

Predlogi

Kar se tiče izgubnega datotečnega formata JPEG, lahko z uporabo primernih nastavitev, dosežemo izredno visoka kompresijska razmerja in visoko kvaliteto slike, brez izrazitejših popačenj. Tukaj svetujem nivo stiskanja z vrednostmi 80 ali 90, saj lahko ti pomenijo že 90 % prihranek prostora, z zanemarljivim popačenjem slike. Strogo pa se je tukaj potrebno izogibati enobarvnih slik. Za stiskanje enostavnih grafik, se format GIF izkaže nekako bolje kakor PNG. Uporabo PNG formata svetujem takrat, kadar imamo opravka z logotipi in prosojnimi slikami.

Za vektorske slike mislim, da je bolj smiselno uporabljati SVGZ format, kakor kakšen arhivsko-kompresijski format. SVGZ namreč že podpirajo nekateri spletni brskalniki, čas, ki je potreben za njegovo odpiranje pa je zanemarljiv. Dodatno stiskanje SVGZ formata v kakšen arhivsko-kompresijski format ni smiselno, je pa smiselno stisniti in zapakirati večje število vektorskih slik za prenos preko spleta.

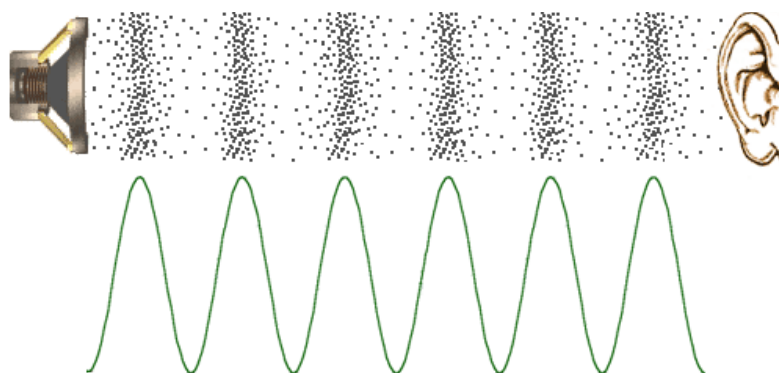
8 ZVOK V SPLETU IN NJEGOVE LASTNOSTI

Še en pomemben element, ki ga pogosto srečamo na spletu je zvok. Ta nam je danes že tako samoumeven, da na trenutke celo pozabimo, kako pogosto ga uporabljamo. Ta večinoma spremlja vse video posnetke na spletu, poživlja spletne animacije, omogoča poslušanje glasbe ali opravljanje klica preko interneta (VoIP) in če smo odkriti, lahko priznamo, da nas je malo takšnih, ki ne bi doma imeli vsaj nekaj s spleta »sposojene« glasbe.

Digitalni zvok je digitalna reprodukcija zvoka z uporabo impulzno kodne modulacije (angl. pulse code modulation - PCM) in digitalnih signalov. Bistvenega pomena so tukaj analogno-digitalni in digitalno-analogni pretvorniki, za kvalitetno procesiranje in prenašanje zvoka ter hitri in enostavni dekodirniki, za njegovo predvajanje. Največja prednost digitalnega zvočnega zapisa je njegovo shranjevanje, prenašanje in kopiranje, brez popačenj ali izgube kvalitete.

Pri stiskanju zvoka za uporabo v spletu, je izbira kodirnika odvisna glede na to, ali se bo njegov prenos izvajal in obdeloval v realnem času (video klici, klici preko interneta, prenosi v živo...) ali se bo prenesel klasično, za poslušanje glasbe, avdio knjig in podobno. Zvok je prav tako, za razliko od slike in teksta, potrebno predvajati, če ga želimo uporabiti.

Zvok je definiran kot mehansko valovanje plina, tekočine ali trdne snovi. Izmed teh je človeško uho najbolj prilagojeno za zaznavanje valovanja zračnega tlaka (stiskanje in raztezanje molekul zraka), katerega kasneje naši možgani interpretirajo kot zvok.



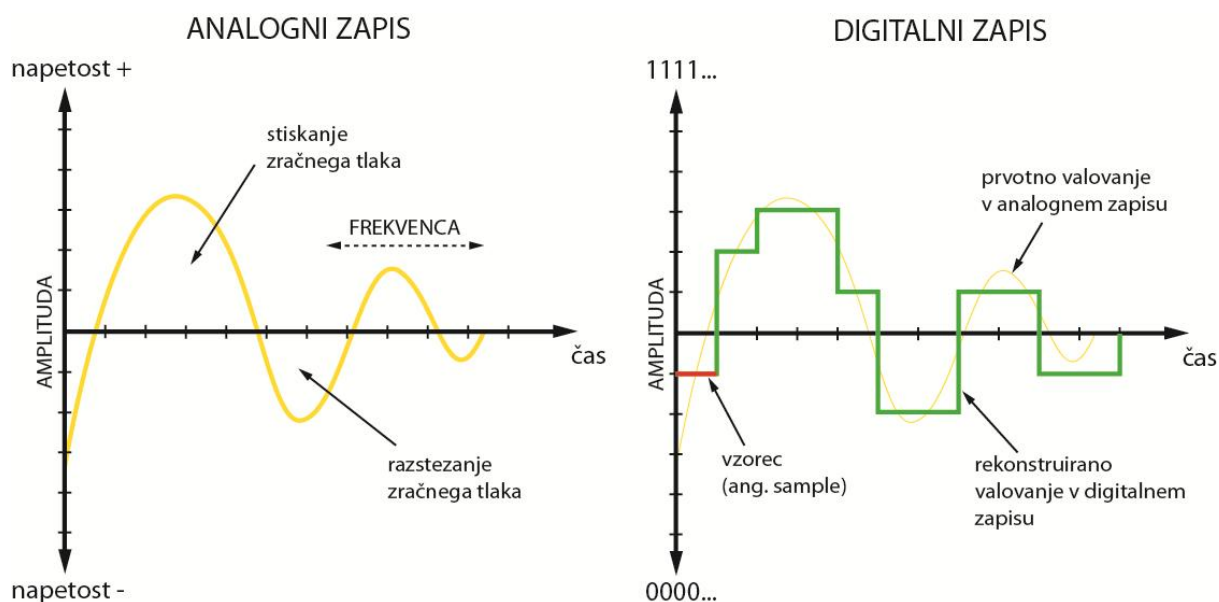
Slika 13: Ponazoritev zvočnega valovanja skozi zrak

Vir: <http://www.mediacollege.com/audio/images/loudspeaker-waveform.gif>, dne 29.8.2011

Kakor sliko, je tudi zvok mogoče predstaviti v obliki ničel in enic. Najpogosteje ga zajamemo z mikrofonom, ki podobno kot uho, zaznava valovanje v zraku in ga spremeni v napetostne vrednosti. Te se spreminjajo skozi čas in predstavljajo analogni zapis zvoka, katerega pa lahko nato pretvorimo v digitalne vrednosti. To početje imenujemo **vzorčenje** (angl. sampling), kjer iz posameznih vzorcev, sestavimo celotno valovanje zvoka, ki predstavlja digitalni zapis zvoka. Iste vzorce, je mogoče pretvoriti nazaj v napetostne vrednosti (analogni zapis), katere nato zaporedno pošiljamo v zvočnik, ta pa nam predvaja zvok. Zvok je lahko generiran tudi s pomočjo programov. Kvalitetna predstavitev zvoka v digitalni obliki, pomeni veliko količino podatkov, zato se za učinkovito stiskanje zvoka, večinoma zahteva tudi dobro poznavanje psihoakustičnih sposobnosti človeškega sluha.

Zvok ima tri pomembne lastnosti: amplitudo, frekvenco in hitrost. Amplituda zvoka predstavlja njegovo jakost, katero merimo v decibelih (dB), človeško uho pa jo zaznava v razmerjih med popolno tišino ali boleče glasnim zvočnim valovanjem. Druga lastnost je frekvenca valovanja, ki je merjena v hercih (Hz) in predstavlja število menjav v valovanju zračnega tlaka. Človeško uho je sposobno slišati frekvence zvoka (slušno področje) nekje od 20 Hz do 20 KHz, pogosteje od 30 Hz do 15 KHz, odvisno od starosti in zdravja osebe. Nižje frekvence od 20 Hz, poznane tudi kot infrazvok ali višje kot 20 KHz, imenovane ultrazvok, človeško uho ne zazna, zato jih ne slišimo. Ostane nam še hitrost, katera pa variira glede na medij, po katerem se zvok prenaša. Zvok potuje hitreje po trdnih snoveh, kakor po plinu. Hitrost zvoka, ki se prenaša preko zraka je okoli 340 m/s, odvisno od temperature zraka. Ta je pomembna za izračun ostalih atributov zračnega valovanja, kateri pa so za to diplomsko nalogo, irelevantni. (Salomon, Motta, 2010, 954)

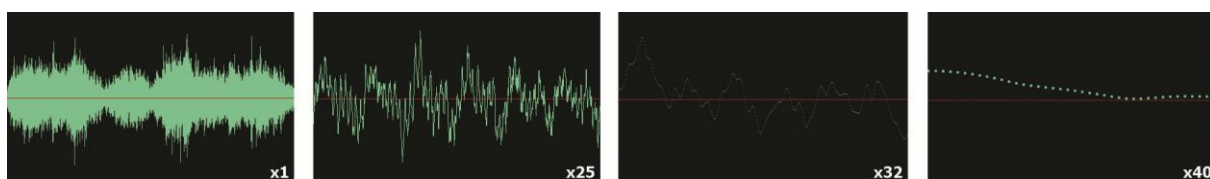
Zvok, katerega zajamemo z mikrofoni je tako ponazorjen kot valovanje amplitude in frekvence v nekem časovnem intervalu. Pri pretvarjanju analognega zvočnega zapisa v digitalnega, gre za pretvarjanje teh, v binarne vrednosti.



Slika 14: Groba ponazoritev pretvorbe iz analognega zvočnega zapisa v digitalnega

Vir: Lasten

Iz zgornje slike je razvidno, da je lahko digitalna predstavitev zvoka veliko slabša od prvotne analogne, saj lahko zaradi premajhnega števila vzorcev pride do precejšnjih izgub podatkov in posledično slabše predstavitve tonov. Za bolj kvalitetno predstavitev zvoka v digitalni obliki, ga je tako potrebno zajeti z večjim številom vzorcev, s katerimi lahko tvorimo natančnejšo predstavitev zvočnega valovanja. Na spodnji sliki lahko vidimo zvočno valovanje predstavljeno s programom Adobe Soundbooth, ki nam z izredno povečavo vzorcev, poda boljšo predstavo o digitalnem zapisu zvoka.



Slika 15: Prikaz digitalnega zvočnega zapisa pri različnih stopnjah povečave

Vir: Lasten

Kvalitetna predstavitev analognega zapisa je tako odvisna od kvalitete pretvarjanja v digitalno obliko in obratno. S tem na kvaliteto digitalnega zvočnega zapisa bistveno vplivata dva parametra:

1. **Stopnja vzorčenja** (angl. sample rate) – število vzorcev (amplitud) na sekundo
2. **Bitna globina** (angl. bit depth) – kako natančno je predstavljena vrednost posamezne amplitude (npr. 8 bitna globina pomeni 256 kombinacij med 0000 0000 in 1111 1111)

Za kvalitetno predstavitev zvoka v digitalni obliki, se s tem poskušamo držati nekaterih standardov. Eden izmed teh je ta, da poskušamo dobiti podobno strukturo zapisa, kakršno uporablja navadni avdio CD. Ta upošteva stopnjo vzorčenja 44.100 Hz, po pravilu Harrya Nyquista, ki trdi, da mora biti stopnja vzorčenja malo nad dvojno vrednostjo najvišje frekvence, če se želimo izogniti slabi kvaliteti zvočnega zapisa. Najvišja vrednost, ki jo premore človeško uho je 20 KHz (20.000 Hz), kar pomeni:

$$20.000 \times 2 = 40.000 + \text{»še malo več«} = 44.100 \text{ Hz (44,1 KHz)}$$

Višje vrednosti stopnje vzorčenja od 44,1 KHz ne dajejo bistvene izboljšave zvoka, medtem, ko jo bitna globina lahko. Bitna globina avdio CD-ja znaša 16 bitov. To pomeni 65.536 možnih kombinacij za predstavitev amplitude zvoka, natančnejša predstavitev amplitude pa pomeni natančnejšo predstavitev zvoka. Sekunda posnetka navadnega avdio CD-ja (44,1 KHz, 16 bit) bo s tem predstavljena sledeče:

$$44.100 \times 16 = 705.600 \text{ bitov} \times 2 \text{ (dva kanala za stereo)} = 1.411.200 \text{ bitov}$$

S tem izračunom vidimo, da sekunda posnetka podane kvalitete, že predstavlja datoteko velikosti 176,4 KB. Uporaba manjših vrednosti lahko povzroči izredno nekvalitetno ponazoritev zvoka in nastanek popačenj. Njihova prekoračitev pa bo s shranjevanjem povzročila izrazito večje datoteke, razlike pa najverjetneje ne bomo slišali. (Salomon, Motta, 2010, 958)

8.1 Stiskanje zvoka

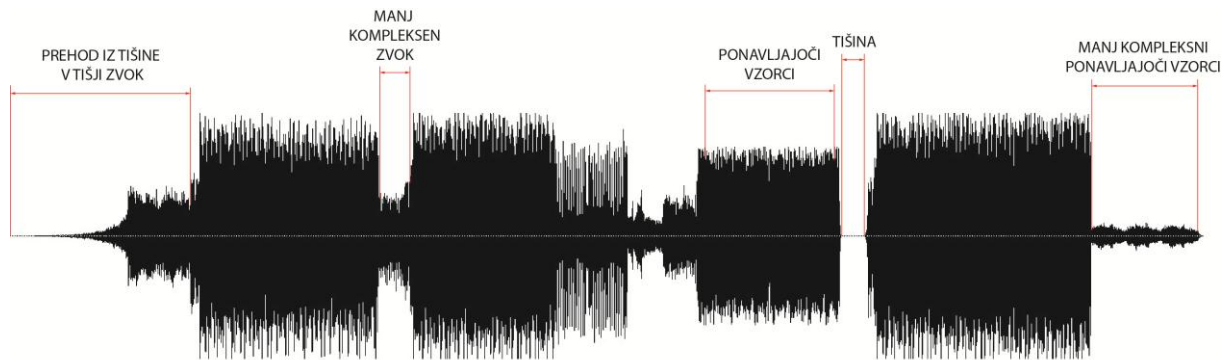
Shranjevanje zvoka v digitalni obliki lahko hitro povzroči nastanek velikih datotek. Različne metode in algoritmi za njegovo stiskanje, se tukaj različno dobro odzivajo na lastnosti zvoka, kot so npr. ponavljajoči se toni, velik delež tišine, umirjena ali bolj živahna glasba. Stiskanje

zvoka se prične s procesom *kvantizacije* (angl. quantization). Kvantizacija je postopek procesiranja zvoka, kjer se analogne vrednosti prilagodijo digitalnim in kjer se načeloma določi primarna kvaliteta digitalnega zvočnega zapisa. Pred procesom kvantizacije, lahko določimo tudi njeno intenzivnost, pri čemer pridemo do pomembnega parametra, imenovanega *bitna hitrost* (angl. bit rate).

Bitna hitrost predstavlja število bitov, ki so preneseni v nekem časovnem obdobju (načeloma eni sekundi). Ta je v primeru zvoka pogosto predstavljena tudi kot *število bitov na vzorec* (angl. bits per sample - bps), ker imamo opravka z vzorci. Ta podatek pride do izraza, kadar začnemo zvočni zapis kodirati in predvajati. Takrat namreč sprožimo pretok podatkov (bitov/vzorcev). V prejšnjem poglavju smo ugotovili, da je sekunda posnetka na avdio CD-ju predstavljena s 1.411.200 biti, kar pomeni, da se med njegovim predvajanjem vsako sekundo prenese 1.411.200 (1411,2 Kbps) nekompresiranih bitov. Ti so nato procesirani in nam predvajani kot zvok. S stiskanjem in predvajanjem kompresiranega zvoka, se nam bo ta vrednost bistveno zmanjšala in spreminjala.

Teoretični primer 1, brez izgubno stiskanje

Brez izgubnega stiskanja se bolj poslužujejo strokovnjaki in audiofili, ker želijo obdržati zvočni zapis v svoji prvotni obliki, brez kakršnihkoli izgub. Tukaj so na delu predvsem razni brez izgubni algoritmi, v kombinaciji z metodami za prepoznavanje vzorcev ali linearno predvidevanje. Ti se po učinkovitosti zelo različno odzivajo glede na zvok, ki ga stiskajo. Tovrstne metode in algoritmi postopoma preverjajo bitne vrednosti zvoka in jih brez izgubno tudi stiskajo. Kadar kodirnik naleti na vzorce tišine, manj kompleksnih zvokov ali ponavljajočih vzorcev, so ti predstavljeni z zaporednimi vrednostnimi ničel ali ponavljajočimi biti. Takšne ponovitve pa je že mogoče stiskati po metodah, ki so bazirane na uporabi slovarja ali celo kakšno enostavno RLE metodo. (Salomon, Motta, 2010, 967)



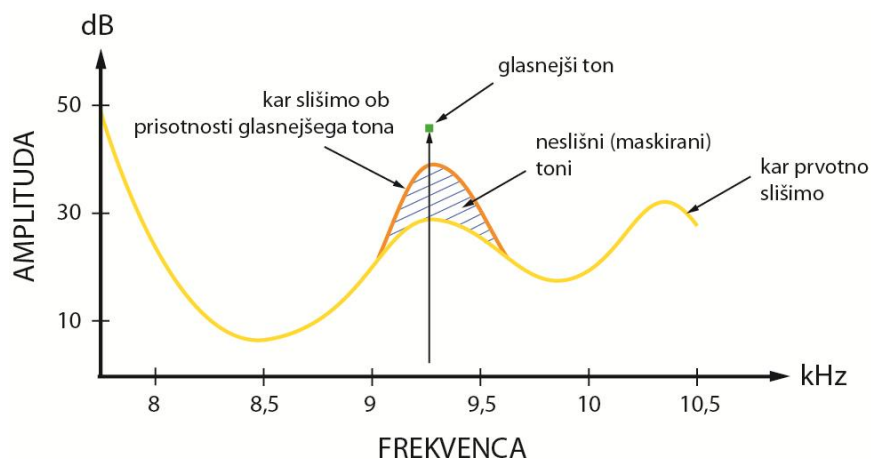
Slika 16: Programsko predstavljeno valovanje zvoka in njegove potencialne možnosti za brez izgubno stiskanje

Vir: Lasten

Teoretični primer 2, izgubno stiskanje

Izgubno stiskanje zvoka temelji na dejstvu, da človeško uho in možgani, niso popolni. Veda, ki se s tem ukvarja se imenuje psihoakustika in veliko prispeva k digitalni obdelavi zvoka. Znano je, da je človeško uho bolj občutljivo na visoke tone (med 2 KHz in 5 KHz), ne sliši zvokov nižjih od 20 Hz in višjih od 20 KHz, ob določenih pogojih ne loči med dvema identičnima tonoma (Haas efekt) in podobno.

Kot primer si pogledajmo izguben način stiskanja zvoka, z uporabo maskiranja. Obstajajo dve vrsti maskiranja: frekvenčno maskiranje (angl. frequency masking) in začasno maskiranje (angl. temporal masking). Za sledeči primer bom uporabil frekvenčno maskiranje. Predstavljajmo si, da poslušamo radio tako na glas, da v tistem trenutku ne slišimo zvonjenje telefona. Z maskiranjem bi tako lahko odstranili zvok zvonjenja, saj ga tako ali tako ne moramo slišati, ker ga radio preglasi. Zvok radia tako prekrije (maskira) zvok zvonjenja in ga naredi neslišnega. Podobno lahko storimo z zvokom podobnih frekvenc. (Salomon, Motta, 2010, 964)



Slika 17: Primer frekvenčnega maskiranja

Vir: Lasten

Gre torej za odstranjevanje zvokov, katerih povprečen poslušalec v teoriji naj ne bi slišal in uporabo dodatnih kodirnih algoritmov ter metod za stiskanje.

S temi principi so nastali izgubni kompresijski zvočni formati, ki nudijo možnost izbire v naprej podanih in standardiziranih bitnih hitrosti. Te se danes gibljejo nekje od 32 Kbps pa do 320 Kbps, kar pomeni prenose od 32.000 do 320.000 bitov na sekundo, med predvajanjem.

8.1.1 Stiskanje govora

Obstajajo tudi metode za stiskanje zvoka, ki so prilagojene predvsem za stiskanje govora. Govor se sicer obravnava enako kot ostali zvočni zapisi, vendar ima ta zaradi narave človeškega glasu nekatere lastnosti, ki jih je mogoče še dodatno izkoristiti v prid stiskanja. Človeški govor se giblje v frekvenčnem območju od 500 Hz do 2 KHz in se zaradi počasnega delovanja glasilk ter pljuč, tudi počasneje spreminja. Pri govoru se načeloma snema samo en govor hkrati in je sneman v mono tehniki (načeloma samo levi kanal). To posledično pomeni tudi manj kompleksen zvok za obdelavo in prenos. Kljub temu je prenos zvoka zahteven postopek, ker zahteva prenose v realnem času. Tukaj se gibljejo stopnje vzorčenja od 8 KHz (ozkopasovne povezave) do 16 KHz (širokopasovne povezave) in redko tudi do 32 KHz (ultra-širokopasovne povezave), z bitnimi hitrosti 2 - 64 Kbps. (Salomon, Motta, 2010, 986)

Kodeki, ki se danes pogosto uporabljajo za stiskanje zvoka so Speex, WMA Voice, SILK (Skype) in razne G.771 ali G.722 variante, za VoIP in digitalno telefonijo. Tovrstni kodeki so v primerjavi s kodirniki kompleksnejšega zvoka, manj zahtevni in kompleksni. Kljub temu,

jih je večina namenjena bolj razvijalcem programske opreme, kot pa širši javnosti. Veliko jih je varovanih kot patenti podjetja.

8.2 Razširjeni zvočni formati

8.2.1 MP3

Izredno popularen in še vedno dominantni datotečni format za shranjevanje zvoka v stisnjeni obliki, je MP3. Razvili so ga pri MPEG, kot del njihovih MPEG-1 in kasneje MPEG-2 standardov. Odobrili so ga leta 1991, javno pa izdali 1993. Gre za patentiran format s kompleksnim kompresijskim algoritmom, ki je sposoben stisniti tipični zvočni zapis avdio CD kvalitete, za desetino njegove prvotne velikosti. Ta podpira le dva kanala (stereo), z možnostjo stopnje vzorčenja 16 - 48 KHz. Njegova kvaliteta zvoka je zadovoljiva večini populacije. Njegova globalna popularnost zagotavlja skoraj popolno kompatibilnost z vsemi napravami in aplikacijami, kar ga naredi idealnega za uporabo v spletu.

(<http://en.wikipedia.org/wiki/MP3>, dne 25.7.2011)

- Polno ime: MPEG-1/MPEG-2 Audio Layer III
- Končnica datoteke: .mp3
- Lastnosti: izguben, zaprt format

8.2.2 AAC

AAC je standardiziran izgubni datotečni format, ki je bil razvit leta 1997, kot naslednik popularnega MP3 formata. Prav tako razvit od skupine MPEG po zgledu MP3 formata, vendar s pomočjo velikih podjetij kot so AT&T, Bell Laboratories, Fraunhofer IIS, Dolby Laboratories, Sony in Nokia. Ta je zasnovan po novejših MPEG-2 in MPEG-4 standardih in je sposoben boljše kvalitete zvoka, pri isti bitni hitrosti kakor MP3. Podpira večje število kanalov (do 48 kanalov) ter višje stopnje vzorčenja (8 - 96 KHz). Je privzeti standardni datotečni format za naprave iPod, iPhone, iPad, PlayStation 3, mobilne telefone z operacijskimi sistemi Android, Symbian, WebOS in podobnimi, katerih uporaba hitro raste in predstavlja vedno večjo grožnjo »zasidranemu« MP3 formatu. Poleg privzetega formata s končnico .aac, je AAC format mogoče uporabiti tudi z drugimi kontejner formati. Obstajajo tudi dve razširitvi AAC kodirnika, z imenom HE-AAC in HE-AAC v2, ki omogočata še dodatne funkcije in izboljšave. Prvoten kodirnik se je ob tem preimenoval v LC-AAC. (http://en.wikipedia.org/wiki/Advanced_Audio_Coding, dne 25.7.2011)

- Polno ime: Advanced Audio Coding
- Končnica datoteke: .aac, .mp4, .m4a, .m4b, .m4p, .m4v, .m4r, .3gp
- Lastnosti: izguben, zaprt format

8.2.3 WMA

Windows Media Audio (WMA), je ime datotečnega formata ali paketa avdio kodekov, ki sta delo podjetja Microsoft. Prvotna verzija tega formata je bila izdana leta 1999, kateri je še isto leto sledila malce izboljšana različica imenovana WMA 2. Kljub temu je zaradi kompatibilnosti dekodiranja med njima, ime ostalo enako. Microsoft sicer trdi, da je WMA sposoben boljše kvalitete zvoka kakor MP3, vendar se ta trditev s strani uporabnikov ni nikoli potrdila. Uporaba in podpora WMA formata je kljub temu na spletu in med različnimi multimedijskimi napravami, zelo razširjena. Paket avdio kodekov WMA, velikokrat spremljajo še trije dodatni kodeki, poznani kot WMA Pro, WMA Lossless, in WMA Voice. (http://en.wikipedia.org/wiki/Windows_Media_Audio, dne 26.7.2011)

- Polno ime: Windows Media Audio
- Končnica datoteke: .wma
- Lastnosti: izguben, zaprt format

8.2.4 Vorbis

Vorbis predstavlja brezplačno in odprto alternativo zgoraj naštetim formatom. Ta se najpogosteje uporablja z Ogg kontejnerjem, zato se ta pogosto imenuje kar Ogg Vorbis. Je delo Xiph.org Fundacije, katerega so začeli razvijati že leta 1993, intenzivno so se ga pa lotili šele 1998. To je privedlo do tega, da je format resnično zaživel šele po letu 2000. Vorbis je sposoben vsebovati 255 kanalov, podpira stopnje vzorčenja 8 - 192 KHz ter bitne hitrosti od 32 do 500 Kbps. Njegova fleksibilnost, kvaliteta in predvsem odprta narava, ga počasi dvigujejo med ostale popularne in razširjene formate. (<http://en.wikipedia.org/wiki/Vorbis>, dne 26.7.2011)

- Polno ime: Vorbis (Ogg Vorbis)
- Končnica datoteke: .ogg, .oga
- Lastnosti: izguben, odprt format

8.2.5 *FLAC*

Vedno bolj priljubljen FLAC format je prav tako delo Xiph.org Fundacije. Ta je poznan po visoki kvaliteti zvoka, ki jo nudi, učinkoviti in brez izgubni metodi stiskanja ter popolni odprtosti uporabe. FLAC je bil razvit okoli leta 2001. Ni sicer tako razširjen in podprt kakor MP3 ali AAC formata, je pa na seznamu bolj razširjenih brez izgubnih avdio formatov. FLAC je sposoben stisniti zvok s prihranki prostora od 30 do 50 %. Podpira bitne globine 4 - 32 bitov na vzorec in stopnjo vzorčenja od enega pa vse do 1.048.570 Hz. Sposoben je vsebovati do 8 zvočnih kanalov. (<http://en.wikipedia.org/wiki/Flac>, dne 26.7.2011)

- Polno ime: Free Lossless Audio Codec
- Končnica datoteke: .flac
- Lastnosti: brez izgub, odprt format

8.3 *Učinkovitost stiskanja zvočnih datotek*

Za sledeče preizkuse učinkovitosti, bom tokrat uporabil dve skladbe različnih zvrsti, a podobnih dolžin trajanja. Za dodaten preizkus bom uporabil še en posnetek govora.

1. **Aktivna propaganda – Here we are.wav**

Vir: avdio CD | stopnja vzorčenja: 44,1 KHz | bitna globina: 16 | zvrst: punk / hardcore | trajanje: 4:12

2. **Electric Boogie – First Source.wav**

Vir: avdio CD | stopnja vzorčenja: 44,1 KHz | bitna globina: 16 | zvrst: elektronika / funk | trajanje: 4:13

Obe skladbi so zajete z originalnih avdio CD-jev v nekompresiranem WAV (Waveform Audio File Format) formatu in stereo tehniki. Temu bom dodal še govor, katerega bom posnel z mikrofonom in kasneje uporabil za stiskanje govora. Zajet govor je: »*Učinkovitost stiskanja tekstovnih, slikovnih, zvočnih in video vsebin za uporabo v spletu*« in je sneman v mono tehniki.

3. **Test govora.wav**

Vir: mikrofoni | stopnja vzorčenja: 44,1 KHz | bitna globina: 16 | zvrst: govor | trajanje: 0:07

Uporabljeni programi

Za zajemanje skladb z avdio CD-jev in pretvarjanje med formati bom uporabil program *Easy CD-DA Extractor* od podjetja Poikosoft in *Adobe Soundbooth* za snemanje govora z mikrofonom. Za naprednejše nastavitve in stiskanje govora pa bom uporabil brezplačni

AudioCoder program. Kot brezplačno alternativo bi tukaj uporabil programa *Audiograbber* za zajemanje in *Helium Audio Converter*, za pretvarjanje med formati. Za snemanje zvoka priporočam *Free Audio Recorder* ali brezplačni večplatformni urejevalnik in snemalnik zvoka, *Audacity*.

Praktični primer 1, brez izgubno stiskanje

DATOTEKA	VELIKOST PRED STISKANJEM	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
Here we are.wav	42,5 MB (44.630.424 B)	32,5 MB (34.118.107 B)	0,76 [24 %]
First Source.wav	42,6 MB (44.715.182 B)	25,6 MB (26.870.973 B)	0,60 [40 %]

Tabela 8: Stiskanje v brez izgubnem FLAC formatu

Vir: Lasten

Praktični primer 2, izgubno stiskanje

DATOTEKA: Aktivna propaganda – Here we are.wav, velikost: 42,5 MB (44.630.424 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
MP3	320 Kbps	9,65 MB (10.122.127 B)	0,23 [77 %]
	192 Kbps	5,79 MB (6.073.357 B)	0,14 [86 %]
AAC	320 Kbps	9,65 MB (10.121.316 B)	0,23 [77 %]
	192 Kbps	5,79 MB (6.072.871 B)	0,14 [86 %]
WMA	320 Kbps	9,69 MB (10.161.717 B)	0,23 [77 %]
	192 Kbps	5,82 MB (6.106.839 B)	0,14 [86 %]
Ogg Vorbis	320 Kbps	9,75 MB (10.230.250 B)	0,23 [77 %]
	192 Kbps	5,85 MB (6.139.269 B)	0,14 [86 %]
DATOTEKA: Electric boogie – First source.wav, velikost: 42,6 MB (44.715.182 B)			
MP3	320 Kbps	9,67 MB (10.142.036 B)	0,23 [77 %]
	192 Kbps	5,80 MB (6.085.324 B)	0,14 [86 %]
AAC	320 Kbps	9,66 MB (10.139.948 B)	0,23 [77 %]
	192 Kbps	5,80 MB (6.084.158 B)	0,14 [86 %]
WMA	320 Kbps	9,74 MB (10.221.353 B)	0,23 [77 %]
	192 Kbps	5,85 MB (6.143.769 B)	0,14 [86 %]
Ogg Vorbis	320 Kbps	9,78 MB (10.255.597 B)	0,23 [77 %]
	192 Kbps	5,87 MB (6.155.880 B)	0,14 [86 %]

Tabela 9: Preizkus stiskanja s formati MP3, AAC, WMA in Vorbis

Vir: Lasten

Praktični primer 3, izgubno stiskanje v VBR načinu

Vsi dosednji preizkusi stiskanja so bili izvedeni v načinu konstantne bitne hitrosti (angl. constant bit rate – CBR), obstaja pa tudi možnost tako imenovane variabilne bitne hitrosti

(angl. variable bit rate – VBR), kjer se bitna hitrost, ki se uporablja za kodiranje, prilagodi glede na intenzivnost in kompleksnost vhodnega toka podatkov. S tem načinom bi naj bilo mogoče dobiti najboljše razmerje, med kakovostjo in velikostjo zvočnega (ali video) zapisa. Naredimo preizkus z izredno udarno skladbo skupine Aktivna propaganda in razgibanimi ritmi Electric boogie-a.

DATOTEKA: Aktivna propaganda – Here we are.wav, velikost: 42,5 MB (44.630.424 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
MP3	~ 320 Kbps VBR	8,70 MB (9.127.167 B)	0,20 [80 %]
	~ 192 Kbps VBR	5,24 MB (5.495.335 B)	0,12 [88 %]
AAC	~ 320 Kbps VBR	9,50 MB (9.962.754 B)	0,22 [78 %]
	~ 192 Kbps VBR	5,73 MB (6.015.882 B)	0,13 [87 %]
WMA	~ 320 Kbps VBR	9,86 MB (10.341.159 B)	0,23 [77 %]
	~ 192 Kbps VBR	5,40 MB (5.670.603 B)	0,13 [87 %]
Ogg Vorbis	~ 320 Kbps VBR	10,4 MB (11.000.715 B)	0,25 [75 %]
	~ 192 Kbps VBR	6,08 MB (6.376.756 B)	0,14 [86 %]
DATOTEKA: Electric boogie – First source.wav, velikost: 42,6 MB (44.715.182 B)			
MP3	~ 320 Kbps VBR	7,43 MB (7.799.057 B)	0,17 [83 %]
	~ 192 Kbps VBR	5,46 MB (5.730.731 B)	0,13 [87 %]
AAC	~ 320 Kbps VBR	9,70 MB (10.181.091 B)	0,23 [77 %]
	~ 192 Kbps VBR	5,86 MB (6.145.213 B)	0,14 [86 %]
WMA	~ 320 Kbps VBR	8,13 MB (8.528.803 B)	0,19 [81 %]
	~ 192 Kbps VBR	4,72 MB (4.954.835 B)	0,11 [89 %]
Ogg Vorbis	~ 320 Kbps VBR	8,39 MB (8.805.086 B)	0,20 [80 %]
	~ 192 Kbps VBR	4,95 MB (5.194.201 B)	0,12 [88 %]

Tabela 10: Stiskanje zvočnih datotek v VBR načinu

Vir: Lasten

Praktični primer 4, izgubno stiskanje govora

Stiskanje govora je sicer mogoče z običajnimi kodirniki, kot so MP3, AAC, WMA in Vorbis, vendar so manj primerni za hitro pretakanje zvoka pri video klicih, klicih preko spleta ali VoIP tehnologijah. Tovrstni kodeki so manj kompleksni, ampak strogo prilagojeni za stiskanje glasu, katerih rezultat so majhne stopnje vzorčenja z solidno kvaliteto glasu. Kodirniki glasu kot so Speex, SILK (uporabljen pri Skype) ali G.711 variante (mobilne telefonije, VoIP) so temu primernejši, ampak pri tem uporabljajo tudi manj podprte formate. Ti se zato načeloma uporabljajo v sklopu svojih aplikacij in ne za tipične prenose preko interneta.

Nekaj tovrstnih kodirnikov imam sicer na voljo, nimam pa primernih dekodirnikov in predvajalnikov. Kljub temu, bom v tej diplomski nalogi še za lažjo predstavo, posebej preizkusil nekatere formate. Pri tem sta formata MP3 in WMA Voice, edina opravila preizkus brez težav. Ostali (PCM, Speex) so zaradi programskih in strojnih omejitev, povzročali težave.

DATOTEKA: Test govora.wav, velikost: 619 KB (634.834 B)			
UPORABLJEN FORMAT	KAKOVOST STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
MP3	16 Kbps	14,2 KB (14.548 B)	0,02 [98 %]
WMA Voice	16 Kbps	21,8 KB (22.326 B)	0,04 [96 %]

Tabela 11: Stiskanje govora pri 16 Kbps (širokopasovne povezave)

Vir: Lasten

8.4 Ugotovitve

Največji problem, ki se pojavi z ocenjevanjem kvalitete pri digitalnem zvočnem zapisu, je zahtevnost samega poslušalca. Veliko ljudi danes brska po spletu s prenosniki, kateri imajo v sebi vgrajene zvočnike nizkih kakovosti in slabih zvočnih sposobnosti. Ti uporabniki po tem takem ne morejo slišati razlike med zvokom, ki je stisnjen kakovostno ali tistim, ki ni. Na drugi strani pa imamo oboževalce kvalitete in avdiofile, ki uporabljajo kakovostne zvočne sisteme za predvajanje ter poslušanje zvoka in seveda ne želijo uporabljati nekvalitetno stisnjenih zvočnih datotek. Kako torej zadovoljiti obe strani?

FLAC je sposoben brez izgubno stisniti zvočne datoteke tudi za polovico svoje prvotne velikosti. Učinkovitost njegovega stiskanja je sicer odvisna od kompleksnosti zvoka, a je kljub temu sposoben v povprečju, s stiskanjem privarčevati 30 – 50 % v velikosti datoteke. Velikosti brez izgubno stisnjenih datotek pa so še vseeno prevelike, za pretakanje ali prenose večjega števila zvočnih datotek preko interneta. Tukaj se zato pogosteje uporabljajo izgubni formati. Stiskanje zvoka z uporabo formatov MP3, AAC, WMA ali Vorbis, je za tovrstno uporabo zaradi visokega kompresijskega razmerja, učinkovitejša. Vsi ti formati prav tako podpirajo pretakanje. S preizkusi sem tukaj ugotovil, da dosežejo vsi testirani zvočni formati z istimi parametri stiskanja, isto kompresijsko razmerje (ne pa velikosti datoteke). Zaradi presenetljivih rezultatov, sem tukaj stiskanje izvedel še z drugimi programi (AudioCoder, Helium Audio Converter), da se prepričam o verodostojnosti meritev, ampak rezultati so ostali nespremenjeni. Ista zgodba se pojavlja tudi drugje po spletu (npr.

<http://nigelcoldwell.co.uk/audio/> , dne 30.7.2011). Torej, ne glede na to ali uporabimo format MP3, AAC, WMA ali Vorbis, kompresijsko razmerje pri stiskanju s konstantnimi bitnimi hitrostmi, ostaja enako (0,23 pri 320 Kbps in 0,14 pri 192 Kbps). Isto velja za stiskanje govora. Med stiskanjem je bil Ogg Vorbis malce počasnejši od ostalih. Razlike glede velikosti stisnjenih datotek so zanemarljivo majhne, razliko v kvaliteti posnetkov pa kljub uporabi solidnega ozvočenja, jaz osebno nisem slišal. Zaradi tega, je najverjetneje skozi vsa ta leta MP3 format, še vedno na prvem mestu, saj večina ljudi razlike med njim in novejšimi formati, za katere trdijo, da nudijo kvalitetnejši zvok, ne slišijo. S stiskanjem zvoka pri uporabi variabilne bitne hitrosti (VBR), prav tako nisem opazil razlike v kvaliteti zvoka. Sem pa opazil, da je s tem mogoče na račun malo daljšega časa pri stiskanju, dobiti še boljša kompresijska razmerja. Ob tem se je dobro prepričati glede podprtosti variabilne bitne hitrosti, s strani kompresijskih programov in naših predvajalnikov, ker te ne podpirajo vsi.

S stiskanjem govora, je postalo razvidno, da ga je mogoče prav tako učinkovito stisniti, vendar moramo biti pozorni na kvaliteto stisnjenega glasu. Tukaj je sicer MP3 nudil boljšo kompresijsko razmerje, ampak ga je WMA Voice premagal, z bistveno boljšo kvaliteto zvoka.

Predlogi

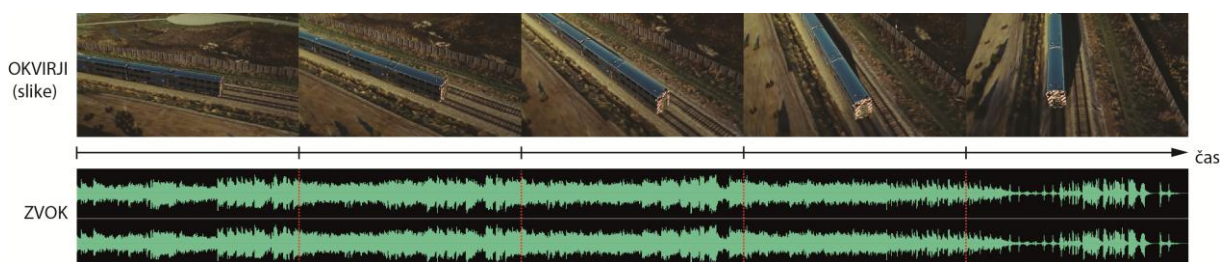
Za privržence kvalitete, arhiviranje zvočnih datotek ali strokovnjake na tem področju, je zvočni datotečni format FLAC, prava rešitev. Kar se pa tiče izgubnih zvočnih formatov, se mi zdi AAC primernejši, kadar želimo tisti dodaten kanček kvalitete; MP3 pa nam bo odlično služil, kadar želimo glasbo poslušati kjerkoli in kadarkoli. Tukaj lahko rečemo, da gre pri tej izbiri tudi za stvar okusa. Preizkus stiskanja je pokazal, da nam vsi popularni izgubni formati, pri uporabi istih parametrov stiskanja, dajo isto kompresijsko razmerje. To naredi formate na področju varčevanja prostora enakovredne. Problem, ki se ob tem pojavi je kvaliteta zvoka, ki jo pri tem nudi vsak format, saj se razlikuje od poslušalca do poslušalca. Tukaj se zna zmogljiv človeški um včasih poigrati z nami tako, da slišimo tisto, kar želimo slišati. Zaradi tega tukaj priporočam, da poslušalcu ponudimo tisto, kar želi slišati. To storimo tako, da izdelamo več stisnjenih datotek, različnih formatov in kakovosti, uporabnik pa lahko nato svobodno izbira, katera bo najboljše ustrezala njegovemu ušesu.

Dober primer je spletna stran bandcamp.com.

9 VIDEO V SPLETU IN NJEGOVE LASTNOSTI

Video nam podaja informacije tako vizualno kot slušno. To nam predstavlja zelo naraven in učinkovit način podajanja informacij. Njegova raba v spletu, raste iz dneva v dan in zahteva razvoj vedno večjih pasovnih širin ter kapacitet medijev, za njegovo hranjenje. Njegov prenos preko interneta nam omogoča pretočne video vsebine, video klice, video konference ter podobne. Veliko prometa prav tako ustvarjajo piratske video vsebine (nelegalno pridobljeni filmi, nadaljevanke, oddaje, ...).

Video je mogoče zajeti z digitalnimi video kamerami, ga pretvoriti iz analognega zapisa v digitalnega ali pa programsko ustvariti s pomočjo računalnika, v obliki animacije. Ta predstavlja sekvenco zaporednih slik v nekem časovnem intervalu, katerega večinoma spremlja še zvok. Kadar snemamo z video kamero, nam ta zelo hitro izdeluje zaporedja slik (na primer 25 slik v eni sekundi), istočasno pa snemamo tudi zvok. Posamezna slika v videu se imenuje okvir (angl. frame), ki sestavlja zaporedje oziroma sekvenco okvirjev. Te okvirje nato prikazujemo enega za drugim z določeno hitrostjo, s čimer ustvarjamo iluzijo gibanja in animacije. Ob tem načeloma še istočasno predvajamo zvok, ki je z njimi časovno usklajen. Ta zaporedja slik in zvoka, prav tako spremljajo lastnosti kot so ločljivost, barvna globina ali stopnja vzorčenja in prav tako vplivajo na kvaliteto videa in velikost datoteke; vendar jih je, kakor smo spoznali iz prejšnjih poglavij, prav tako mogoče stisniti in optimizirati.



Slika 18: Video kot kombinacija zaporednih slik in zvoka

Vir: Source Code, 2011; lasten

Slike oziroma okvirji v videu, so kakor rastrske slike, prav tako sestavljeni iz x vrstic (horizontalnih linij) in y stolpcev (vertikalnih linij) barvnih pik, ki kot celota sestavljajo eno sliko z ločljivostjo $x \times y$. Ločljivost okvirjev je prirejena glede na sistem, na katerem se bo video predvajal (televizija, kino, DVD, Blu-Ray, računalniški zaslon, ...). Te zaslonske ločljivosti se od sistema do sistema spreminjajo, zato se tukaj uporablja tako imenovano

razmerje slike (angl. aspect ratio), ki določa primerno širino in višino ($x : y$) slike, ne glede na njeno ločljivost. Barvni modeli, ki se tukaj uporabljajo, so večinoma bazirani na barvnem modelu RGB in YUV. Slike se v videu izrisujejo v dveh načinih: s *prepletanjem* (angl. interlaced) ali *progresivno* (angl. progressive). Prepletanje je stara tehnika, ki predstavlja prvi pristop k stiskanju videa in to tako, da se posamezen okvir najprej izriše iz lihih horizontalnih vrstic, šele nato pa sledi izris sodih horizontalnih vrstic. Ta tehnika omogoča dobro kvaliteto slike, tudi pri uporabi nizkih pasovnih širin. Progresivni izris slike pa predstavlja nasprotje prepletanja, s tem, da izriše vse vrstice hkrati. Progresivni način se danes uporablja že v vseh računalniških monitorjih ter pri večini plazma in LCD televizorjih, zato je video na spletu prav tako podan tudi v tej tehniki. V nasprotnem primeru, mora biti ta pretvorjen v progresivno obliko (postopek pretvorbe se v angleščini imenuje deinterlacing), sicer bomo priča neprimernemu izrisu okvirjev. Iluzija gibanja in animacije nastane z zaporednim izrisovanjem okvirjev na ekranu, enega za drugim. Človeško oko zaradi svoje vztrajnosti ne zazna zelo hitrih sprememb slike, zato nastane iluzija gibanja že pri 15 okvirjih na sekundo. Okvirji na sekundo (angl. frames per second – **FPS**), je meritev, ki nam pove koliko okvirjev se je zaporedno izrisalo v eni sekundi ter se prav tako uporablja v računalniški grafiki in video kamerah. Za nemoteno prikazovanje gibov so ti določeni po standardih: 23,976 ali 24 FPS (kino), 25 FPS ali 30 FPS (televizija) ter 50 ali 60 FPS za video visoke ločljivosti (HD, Blu-Ray) itd. S številom okvirjev na sekundo lahko predstavimo tudi *hitrost osveževanja*, ki predstavlja število slik, ki jih je ekran sposoben izrisati v sekundi. Ta se meri v hercih (Hz) in se pogosteje uporablja pri strojni opremi, medtem, ko je sorodni FPS, bolj programske narave. Celotno tovrstno sekvenco okvirjev pa načeloma spremlja še zvok. Veliko modernih filmov je danes posnetih v večkanalni tehniki zvoka, ki nam z uporabo pravega zvočnega sistema, zna pričarati občutek prostorskega zvoka. Vse dodatne izboljšave s strani zvoka pa prav tako prispevajo k velikosti celotnega video zapisa. Pri digitalnem videu se tako vsa ta bitna hitrost zaporednih slik in zvoka, seštevata.

9.1 Video v računalništvu

Sprva je bil video le del televizijske in filmske industrije. Ta je prav tako postavila standarde za njegovo snemanje in predvajanje. Tukaj se lahko lastnosti videa kot so ločljivost, razmerje slike, število okvirjev na sekundo in frekvenca osveževanja, razlikuje glede na standard, ki ga države uporabljajo (NTSC, PAL ali SECAM). S pojavitvijo potrošniških video kamer,

splošno digitalizacijo naprav, LCD televizorjev, Blu-Ray medijev ter drugih multimedijskih naprav pa so se pojavili še drugi standardi. Razlika med televizijskim in računalniškim videom sicer ni velika, vendar obstajajo razlike v standardih in formatih, katerih se je za pravilen prikaz videa, potrebno držati.

S to diplomsko nalogo se želim osredotočiti predvsem video, ki se uporablja na spletu in je prilagojen za zaslone računalniških monitorjev, prenosnikov, pametnih telefonov in ostalih multimedijskih naprav. Njihovi zasloni so različnih ločljivosti, barvne pike, ki jih sestavljajo pa so štirikotnih oblik, medtem, ko so zasloni televizorjev, pravokotnih oblik. To lahko privede do neprimerne velikosti slike. Potrebno je upoštevati razmerje pike (angl. pixel aspect ratio - PAR) in razmerje zaslona (angl. display aspect ratio - DAR). Razmerje pike pri računalniških zaslonih in multimedijskih napravah, je 1.0 (pomeni merilo 1 : 1), razmerje zaslona pa je danes načeloma 16 : 9 (ali 1.78 : 1). (Austerberry, 2005, 78)

Pri določanju kvalitete digitalnega video zapisa, moramo tako biti pozorni na dva parametra, ki sta za njegovo prenašanje preko interneta, bistvenega pomena:

1. **Ločljivost zaslona** (angl. screen resolution) – število vertikalnih in horizontalnih pik, ki sestavljajo okvirje in jih je zaslon sposoben prikazati
2. **Bitna hitrost** (angl. bit rate) – bitno število okvirjev, ki jih vsebuje ena sekunda videa; bitna hitrost videa in zvoka se tukaj seštejeta skupaj

Če poznamo ciljne uporabnike našega videa ali pa enostavno želimo biti kar se le da vsestranski, potem je dobro izbrati temu primerno ločljivost. Tukaj se ločljivosti zaslonov prilagajajo po razmerju slike 4 : 3, 16 : 9 in 16 : 10, izmed katerih je 4 : 3, razmerje starejših zaslonov, 16 : 10 se počasi umika, najpogosteje pa se uporablja 16 : 9.

SISTEM	16 : 9	16 : 10	4 : 3
Računalniški monitorji in zasloni prenosnikov	1920 × 1080	1680 × 1050	1600 × 1200
	1600 × 900	1440 × 900	1280 × 960
	1366 × 768	1280 × 800	1024 × 768
	1280 × 720	1024 × 640	800 × 600
Tablični računalniki	1024 × 600	1280 × 800	1024 × 768
	1024 × 576		800 × 600
	854 × 480		
Mobilni telefoni, dlančniki	640 × 360	768 × 480	640 × 480
		640 × 400	480 × 320
		320 × 200	320 × 240

Tabela 12: Najpogostejša razmerja in ločljivosti zaslonov digitalnih naprav

Vir: http://en.wikipedia.org/wiki/List_of_common_resolutions, dne 5.9.2011

Z izbiro primerne ločljivosti videa in njegove podatkovne hitrosti, lahko dosežemo optimalno razmerje med velikostjo datoteke in kvaliteto videa, za prenose preko interneta. Ločljivost videa je lahko seveda tudi poljubnih dimenzij, dokler se držimo nekega standardnega razmerja slike. Pri iskanju primerne ločljivosti si lahko pomagamo s sledečimi primeri:

- $640 \times 9 / 16 = 360$ - (ločljivost 640×360 , 16 : 9)
- $640 \times 3 / 4 = 480$ - (ločljivost 640×480 , 4 : 3)
- $640 \times 10 / 16 = 400$ - (ločljivost 640×400 , 16 : 10)

Prednost predvajanja videa, na računalniku ali kateri drugi multimedijски napravi je ta, da je za večino dela odgovorna programska oprema. To pomeni, da lahko video predvajamo ne glede na njegovo obliko kodiranja, vse dokler ima predvajalnik na razpolago primeren dekodirnik in podpira datotečni format. Kadar predvajamo video s takšnim predvajalnikom, mora ta hkrati opravljati sledeče:

- interpretirati kontejner format, da pridobi informacije od avdio in video tokovih za njuno dekodiranje;
- dekodirati video tok in pošiljati sekvence slik grafični kartici;
- dekodirati zvokovni tok in ga pošiljati zvočni kartici.

Na operacijske sisteme je mogoče naložiti vrsto video in multimedijских predvajalnikov. Od teh je večina narejenih za predvajanje video datotek, veliko pa jih podpira predvajanje pretočnih vsebin preko spleta. V primeru spletnih brskalnikov, se za te namene danes večinoma uporabljajo tako imenovani vtičniki (angl. plugins). Ti vtičniki služijo kot

predvajalniki, ki so prilagojeni za delovanje s spletnim brskalnikom. Najpogosteje uporabljeni vtičniki za predvajanje pretočnih vsebin, so na primer Flash, QuickTime, Real in Windows Media. (<http://diveintohtml5.org/video.html#divingin>, dne 9.9.2011)

9.2 HTML5

Novost, ki počasi prodira na splet, je implementacija novega označevalnega jezika HTML5 (angl. Hyper Text Markup Language revision 5). HTML5 po novem omogoča uporabo elementa `<video>` za enostavno vdelavo videa, v spletno stran. Ta element sedaj omogoča povezavo do ene ali več video datotek, katero bo spletni brskalnik avtomatsko predvajal, z uporabo vgrajenega dekodirnika. HTML5 bo s tem močno poenostavil uporabo videa na spletu, saj bo video v spletnem brskalniku mogoče predvajati brez vtičnikov, istočasno pa se bo povečala kompatibilnost med video vsebinami, spletnimi brskalniki in multimedijskimi napravami. (<http://diveintohtml5.org/video.html#divingin>, dne 9.9.2011)

9.3 MPEG-4

MPEG-4 je standard / način podajanja kompresijskih metod, za stiskanje zvočnih in vizualnih digitalnih vsebin. Ta se je kot projekt začel maja 1991 ter bil predstavljen in določen kot standard, leta 1998. Ta standard, zbirke zvočnih in video kodirnikov ter formatov, so določili pri ISO/IEC in MPEG, pod imenom ISO/IEC 14496 – kodiranje avdio-vizualnih objektov. Ta posega v vsa področja avdio-vizualne produkcije, distribucije in potrošnje na področju računalništva, televizije, glasbene produkcije, telefonije, radija, spleta in podobnih. Ime MPEG-4 lahko s tem povzroča precej zmede, saj se isto ime uporablja za podajanje kodekov (npr. MPEG-1, MPEG-2, MPEG-4, ...), kontejner formata (MP4) in zbirko standardiziranih kompresijskih metod; zato bodimo pozorni. (<http://en.wikipedia.org/wiki/MPEG-4>, dne 10.9.2011)

9.4 Stiskanje videa

Kvaliteten digitalni video zapis, predstavlja izredno veliko količino podatkov za njegovo predstavitev, zato se je na spletu začel masovno pojavljati šele v zadnjih nekaj letih. Zaradi njegove obsežnosti, je ta že od nekdaj deležen stiskanja. Tukaj je potrebno upoštevati elemente slike in zvoka ter njiju pomnožiti s časom.

Predstavljajmo si, da imamo nek povprečen digitalni fotoaparati, ki je sposoben snemati video in z njim naredimo kratek 30 sekundni video posnetek. Ta najverjetneje omeji velikost slike na ločljivost 640×480 . Slike bo zajemal s hitrostjo 30 sličic na sekundo, zraven pa bo preko vgrajenega mikrofona snemal še en kanal zvoka. Brez stiskanja, bi video na fotoaparatu zasedel sledečo velikost:

- *Slika / okvir:* $640 \times 480 \times 24 = \underline{7.372.800 \text{ bitov}} \text{ (921 KB)}$
- *Število slik v eni sekundi:* $7.372.800 \times 30 = \underline{221.184.000 \text{ bitov}} \text{ (27 MB/s)}$
- *Sekvenca slik v času 30 sekund:* $7.372.800 \times 30 = \underline{6.635.520.000 \text{ bitov}} \text{ (810 MB)}$

Ne pozabimo na zvok! Vgrajen mikrofoni je povprečne kakovosti in je sposoben zajeti zvok, pri stopnji vzorčenja 32 KHz in 16 bito globino.

- *Sekunda zvoka:* $32.000 \times 16 = \underline{512.000 \text{ bitov}} \text{ (64 KB/s)}$
- *30 sekund zvoka* $= 512.000 \times 30 = \underline{15.360.000 \text{ bitov}} \text{ (1.8 MB)}$

Teh 30 sekund nekompresiranega posnetka s fotoaparata, bi tako zavzelo:

$$6.635.520.000 \text{ b (810 MB)} + 15.360.000 \text{ b (1.8 MB)} = 6.650.880.000 \text{ b (811 MB)}$$

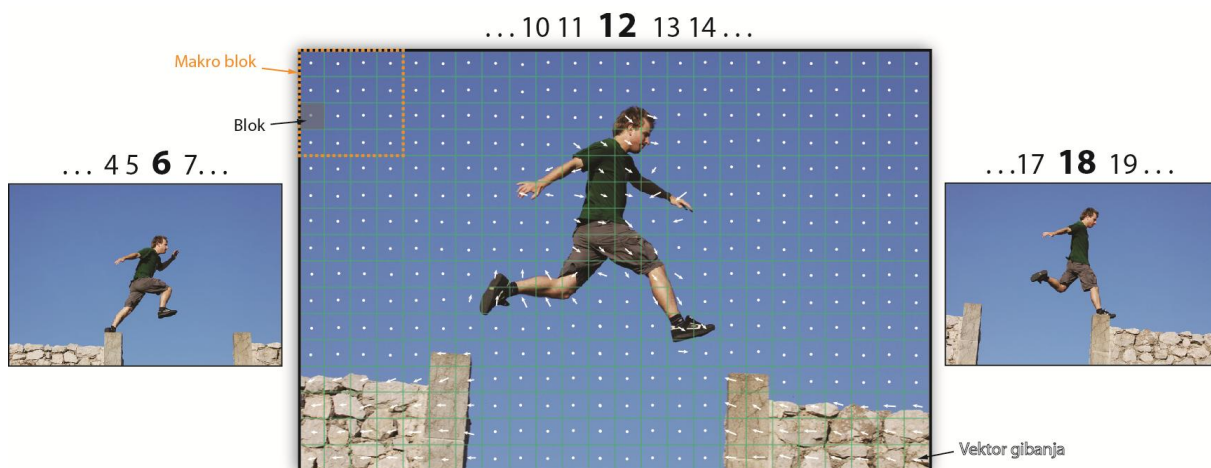
S takšnim 30 sekundnim video posnetkom, bi na fotoaparatu tako izgubili prostora za približno 270 fotografij.

V primeru televizijskih sistemov standardne (SD) ali visoke ločljivosti (HD), zavzame dve ure nekompresiranega digitalnega videa, od 110 do 670 GB prostora in prenose od 125 do 750 mbps. Medtem, ko je danes tipičen DVD sposoben hraniti 4,7 GB podatkov, pretakanje videa pa se načeloma izvaja pri prenosih 1 – 15 mbps. S tega je razvidno, da je razvoj stiskanja na tem področju, kritičnega pomena. Za tovrstne standarde, kompresijske algoritme in metode, so danes zadolženi prepoznavni organi za standardizacijo, kot so International Organization for Standardization (ISO), International Telecommunication Union (ITU) ali Motion Picture Expert Group (MPEG). Video se večinoma poslužuje izgubnih načinov stiskanja (še posebej za uporabo v spletu), pri čemer se išče dobro razmerje med njegovo kvaliteto, velikostjo zapisa in strojno zahtevnostjo za njegovo razširitev in predstavitev. (Salomon, Motta, 2010, 867)

Za stiskanje videa je mogoče izkoristiti več različnih metod. Okvirjev, ki sestavljajo video, je ogromno, vendar so razlike med njimi, večinoma časa majhne, kadar primerjamo vsakega s sosednjim. Do večjih sprememb pride le pri menjavi kadra ali intenzivnim spremembam v sceni (npr. akcijska scena v filmu). S tem je mogoče vsak »prvi« okvir stisniti, po metodah tipične mirujoče slike. V okvirjih, ki mu sledijo pa je mogoče stiskati samo razlike, ki se med njimi pojavljajo. Ponavljajoči elementi prvega (predhodnega) okvirja, se tako kopirajo v drugega in v vse sledeče. To metodo lahko kodirnik ponavlja, vse dokler se ne pojavi popolnoma drugačen okvir, katerega je potrebno ponovno stisniti, po metodah mirujoče slike. Okvir, ki se stiska po svojem predhodniku se v angleškem jeziku imenuje *interframe*, medtem, ko se vsak posebej kodiran okvir, v angleščini imenuje *intraframe*. Interframe in intraframe stiskanje, je osnova v stiskanju videa. (Austerberry, 2005, 80)

Teoretični primer 1

Sledeča metoda za stiskanje videa je primer kompenzacije gibanja. Ta metoda spada v sklop interframe kodiranja in predstavlja pomemben element, kadar uporabljamo MPEG kodirnike. Primer, ki ga predstavljam, se natančneje imenuje kompenzacija gibanja z ujemanjem blokov fiksnih velikosti (angl. fixed size block-matching motion compensation). Ta metoda razdeli vsak posamezen okvir (referenčni okvir) v fiksno število makro blokov (večinoma kvadrati velikosti 16×16 , 8×8 ali 4×4 ; lahko so tudi pravokotniki), preko katerih se nato s primerjavo vseh posameznih blokov, v sledečih okvirjih išče podobnosti med njimi. Bloki, ki se dobro ujemajo po podobnosti, se nato uporabijo le iz prvega (referenčnega) okvirja, ostali pa izpustijo. Vsem ostalim blokom, v katerih smo našli razlike (katere je najverjetneje povzročil premik kamere ali objekta) pa se obdelajo in stisnejo. Dodatno se jim dodelijo *gibalni vektorji* (angl. motion vector), ki služijo kodirniku za učinkovitejše iskanje podobnosti in predvidevanje sprememb, v sledečih blokih. Gibalni vektorji prav tako dobro služijo dekodirniku, za lažje in hitrejše dekodiranje podatkov. (<http://www.dcs.warwick.ac.uk/research/mcg/bmmc/index.html>, dne 16.9.2011)

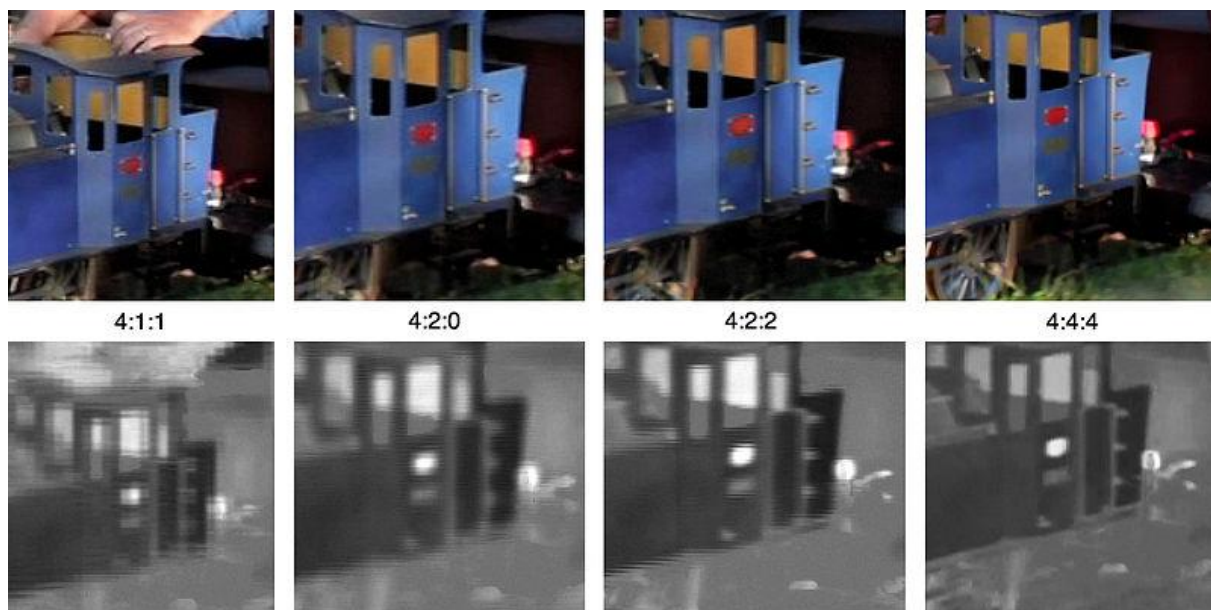


Slika 19: Primer kompenzacije gibanja

Vir: Lasten

Teoretični primer 2

Stiskanje videa se prav tako poslužuje izkoriščanju slabosti človeškega vida. Kakor sem že v poglavju slike podal, je človeško oko bolj občutljivo na spremembe v svetlobi kakor v barvah. V video sistemih, se signal večinoma prenaša po sistemu podajanja luminance in krominance - Y'CbCr. Y'CbCr predstavlja neke vrste videu optimiziran RGB barvni model, ki daje večjo prednost lumna komponenti Y' (podatki o svetlobi), kot pa barvni razliki predstavljeni v kroma komponentah Cb in Cr (podatki o barvah). Tukaj se slika deli na makro bloke velikosti 16 x 16, katere podobno kot pri JPEG formatu, stisnemo po metodah kvantizacije in RLE. S tem je mogoče podatke o barvah zapisati pri nižjih ločljivostih. Opisana metoda se imenuje »podvzorčenje« (angl. subsampling) in se uporablja tako za slike, kot za video.



Slika 20: Primer "podvzorčenja" pri različnih nivojih

Vir: <http://en.wikipedia.org/wiki/File:Colorcomp.jpg>, dne 2.9.2011

Zgornja slika prikazuje razliko med štirimi stopnjami »podvzorčenja«, kjer je mogoče opaziti razliko v ločljivosti barv, s katerimi so predstavljene slike v zgornji vrsti. Te se nam sprva zdijo zelo identične, dokler se ne osredotočimo na podrobnosti. Spodnje štiri sivinske slike pa prikazujejo ločljivost barvnega zapisa, s čimer lažje opazimo popačenja (najintenzivnejše na skrajni levi in brez popačenja na skrajni desni), ki so ob tem nastala. (http://en.wikipedia.org/wiki/Chroma_subsampling , dne 30.8.2011)

S kodiranjem videa je mogoče odstraniti veliko redundantnih podatkov, vendar to v večini primerov, predstavlja strojno zahteven in dolgotrajen proces.

9.5 Razširjeni video formati

Kadar se v računalništvu srečamo z videom, nam je ta večinoma dostavljen z zvokom, skupaj pa sta zapakirana v kontejner. Gre torej za datoteko, ki vsebuje eno ali več tokovnih sledi zvoka in videa, pri čemer sta bila oba v njo bila zapisana po svojem formatu. Format video sledi je vanjo zapisal video kodirnik (npr. H.264), format zvočne sledi pa avdio kodirnik (npr. AAC). Opravka imamo torej z dvema formatoma, ki sta zapisana znotraj datoteke, katera pa prav tako predstavlja neke vrste datotečni format, za shranjevanje avdio in video tokov. Tukaj je veliko govora o video kodekih, saj predstavljajo zelo zmogljivo orodje za obdelavo videa.

Večino kodekov se uporablja v sklopu neke knjižnične zbirke ali profilov. Z drugimi besedami lahko rečemo, da kadar uporabljamo kodek, uporabljamo le enega izmed zbirke kodekov, ki so priloženi zraven. Ti se načeloma ločijo po profilih, kot se npr. VC-1 kodek loči po svojih treh različno zmogljivih profilih, imenovanih simple, main in advanced.

9.5.1 H.264

H.264, je najnovejši video format iz velike družine formatov, ki je postal standardiziran maja 2003, kot odgovor na njegove zastarane predhodnike H.261, H.262 in H.263. S poudarkom na hitrosti, fleksibilnosti in zanesljivosti, predstavlja ta en sam, a efektiven kodek, za uporabo z vsemi multimedijskimi napravami, od telefonov do televizorjev in z delovanjem tako pri ozkopasovnimi kot hitrimi širokopasovnimi povezavami. Njegova uporaba v tako širokem obsegu je mogoča z uporabo v naprej definiranih profilih, ki spreminjajo parametre kodirnika glede na ciljno uporabo. Podpira pretakanje preko interneta, visoke in nizke ločljivosti, napredne metode za stiskanje podatkov in še več. Obstaja tudi popolnoma odprta in samostojna različica H.264 kodirnika imenovana x264, ki prav tako stiska video v H.264 formatu. (<http://en.wikipedia.org/wiki/H.264>, dne 11.9.2011)

- Polno ime: H.264 / AVC (Advanced Video Coding) ali MPEG-4 Part 10
- Lastnosti: izguben z možnostjo brez izgubnega, delno patentiran (razen x264)

9.5.2 Theora

Kompresijski video format, katerega počasi razvija Xiph.org fundacija, je razvit iz lastniškega VP3 kodeka. Ta je poznan kot konkurenca ostalim MPEG-4 part 2 standardom, nekaterim Windows Media Video formatom in RealVideo formatu, kljub nekaterimi pomanjkljivostmi, ki jih vsebuje. Kot večina formatov, tudi ta uporablja »podvzorčenje«, blokovno kompenzacijo gibanja in ostale. Ime Theora izvira od istoimenskega igralca iz oddaje Hax Headroom. (<http://en.wikipedia.org/wiki/Theora>, dne 11.9.2011)

- Polno ime: Theora
- Lastnosti: izguben, odprt

9.5.3 VP8

VP8 je kodek podjetja On2 Technologies, ki je prav tako razvilo kodek VP3, na katerem bazira zgoraj omenjena Theora in njegov predhodnik VP6, ki se privzeto uporablja s Flash Video kontejnerjem. V 2010, je to podjetje kupil Google, ki je kodek tudi popolnoma odprl

javnosti ter ga začel razvijati tudi sam. Ta se načeloma uporablja v kombinaciji z novim WebM kontejnerjem. VP8 je na splošno znan kot dober kodek, sicer večjih podrobnosti o njem še ni bilo zaslediti, je pa znano, da vsega kar se Google »dotakne«, postane slej kot prej aktualno za uporabo. (<http://en.wikipedia.org/wiki/VP8>, dne 11.9.2011)

- Polno ime: VP8
- Lastnosti: izguben, odprt

9.5.4 Xvid

Xvid je kompresijski video format, ki je izpeljan iz MPEG-4 part 2 tehnologije stiskanja. Predstavlja resnega tekmeca DivX Pro formatu, ki je nekoč slavil s svojim DivX kodekom, iz katerega Xvid tudi izhaja. Tukaj je šlo predvsem za zaplete v podjetju DivxNetworks (danes Divx, Inc.). Xvid je enostaven za uporabo, a učinkovit pri svojem delu. Na njega prisegajo tudi nekateri izdelovalci in uporabniki piratskih vsebin. (<http://en.wikipedia.org/wiki/Xvid>, dne 11.9.2011)

- Polno ime: Xvid
- Lastnosti: izguben, odprt

9.5.5 VC-1 / Windows Media Video 9

VC-1 je video kodek, katerega so standardizirali pri SMPTE (Society of Motion Picture and Television Engineers), implementiralo pa ga je podjetje Microsoft, kot Windows Media Video 9. Gre torej za kodek z dvema imenoma. Ta je sposoben visokih (do 2048×1536) in nizkih (do 176×144) ločljivosti, nizkih podatkovnih hitrosti ter uporabe treh profilov. Zaradi možnosti kodiranja tako prepletenih, kot progresivnih video signalov, se ta pogosto uporablja tudi v televizijski industriji in predstavlja resno konkurenco H.264 formatu. Ta se načeloma shranjuje v ASF kontejnerju, kateremu pa se zaradi lažje prepoznavnosti, kadar ta vsebuje video, dodeli končnica *.wmv*. V družini Windows Media Video se lahko srečamo tudi s kodekom WMV 7 in WMV 8. (http://en.wikipedia.org/wiki/Windows_Media_Video, dne 11.9.2011)

- Polno ime: Microsoft VC-1 ali Windows Media Video 9 (WMV 9)
- Lastnosti: izguben, delno patentiran

Priporočam še ogled ostalih video kodekov kot so x264, DivX, Dirac, RealVideo, 3vix, VP7 ali pa mogoče tudi nekaj brez izgubnih kodekov, kot so FFv1, Huffvuv, Lagarith in YULS.

9.6 Razširjeni kontejner formati

KONTEJNER	KONČNICA	PODPRTI VIDEO FORMATI	PODPRTI AUDIO FORMATI
MPEG-4	.mp4, .m4v, .m4a, .m4b, .m4p, .m4r	MPEG-2 in MPEG-4 variante, H.264, VC-1, Dirac in ostali	MP3, AAC, Vorbis
Flash Video	.flv, .f4v, .f4p, .f4a, .f4b	H.264, VP6	MP3, AAC
AVI	.avi	Skoraj vse, problemi z H.264	Skoraj vse, problemi z Vorbis
Ogg	.ogv, .ogx, .ogg, .spx	Theora, Dirac, MNG (animiran PNG) in ostali	Vorbis, FLAC, Speex
Matroska	.mkv, .mk3d, .mka, .mks	Vse	Vse
WebM	.webm	VP8	Vorbis

Tabela 13: Tabela primerjave kontejner formatov

Vir: http://en.wikipedia.org/wiki/Comparison_of_container_formats, dne 8.9.2011

MPEG-4 (MP4)

Danes poznan tudi pod polnim imenom MPEG-4 part 14, je kontejner format, ki bazira na starejšem Applovem QuickTime MOV kontejnerju. Razvijajo ga pri MPEG, kot razširitev MPEG-4 part 12 standarda. Ta je standardiziran, sposoben odpravljanja napak, zaščite podatkov, vgradnje profilov in s svojo fleksibilnostjo, sposoben še dodatnih izboljšav. (http://en.wikipedia.org/wiki/MPEG-4_Part_14, 12.9.2011)

Flash Video (FLV)

Flash Video je kontejner podjetja Adobe (nekoč od podjetja Macromedia), ki ga podpira njihov masovno uporabljen predvajalnik Adobe Flash Player. Ta je posebej prirejen za prenose preko interneta in je podprt s strani Adobe-ovih programov. Predstavlja privzeti kontejner, za vse posnetke, ki so naloženi na YouTube-u.

AVI

Kontejner, katerega je leta 1992 razvil Microsoft pod polnim imenom Audio Video Interleave, je preprost kontejner, ki ne nudi veliko naprednih funkcij, ampak še vedno živi.

Zaradi starosti zna povzročati nekaj težav med novejšimi formati, sicer je pa pogosto v uporabi še v piratskih scenah.

Ogg

Ogg je popolnoma odprt standard, za katerega skrbi Xiph.Org Foundation in je namenjen za uporabo, z ostalimi odprti formati kot so Theora, Vorbis, FLAC, Speex in podobnimi. Je izredno sposoben kontejner format, katerega zaradi njegove »odprtosti«, podpirajo vse večje Linux distribucije, popularni brskalniki in predvajalniki. Večjih problemov prav tako ne povzroča na platformah Windows in Mac OS X. (<http://en.wikipedia.org/wiki/Ogg>, dne 12.9.2011)

Matroska (MKV)

Vsestranski multimedijски kontejner format z imenom Matroska (poimenovan po znani tradicionalni Ruski igrači, ki je pri nas poznana kot babuška), bi naj bil sposoben vsebovati neskončno število tokov videa, zvoka, slik in podnapisov. Gre za odprt format, namenjen shranjevanju vseh splošnih multimedijških vsebin in vsesplošne podpore s strani vseh platform. (<http://en.wikipedia.org/wiki/.mkv>, dne 12.9.2011)

WebM

WebM je novejši kontejner format, ki je namenjen za uporabo z VP8 video formatom, Vorbis formatom za zvok ter HTML5. Gre za odprt kontejner format, katerega razvoj sofinancira Google in je že deležen podpore med spletnimi brskalniki Google Chrome, Mozilla FireFox in Opera. (<http://en.wikipedia.org/wiki/Webm>, dne 12.9.2011)

V tovrstne kontejnerje je mogoče shraniti tudi, slike, animacije in tekst. Priporočam tudi ogled ostalih kontejner formatov, kot so MOV (Apple QuickTime Movie) in ASF (Advanced Systems Format).

9.7 Učinkovitost stiskanja video datotek

Pri stiskanju videa gre danes bolj za pretvarjanje (angl. transcoding) iz enega kompresijskega formata v drugega, kot pa kodiranje surovih podatkov.

Tokrat bom za preizkus uporabil dva posnetka, narejena z video kamero Sony DCR-TRV60E PAL, ki nista zapisana v popolnoma nekompresiranem formatu. Gre za format DV, ki je namenjen za digitalno snemanje videa in je delo vodilnih proizvajalcev video kamer. Tukaj je prvi posnetek sestavljen iz dinamičnih kadrov, kjer je veliko gibanja in premikanja kamere, drugi pa je sestavljen iz ponavljajočih se statičnih kadrov in vsebuje malo gibanja. Oba posnetka sta zajeta brez zvočnega zapisa, ker se želim tukaj osredotočiti predvsem na stiskanje videa.

1. Dinamični kadri

Vir: Sony DCR-TRV60E | format: DV | kontejner: AVI | ločljivost: 720×576 | FPS: 25 | bitna hitrost: 24.4 Mbps | trajanje: 1:00

2. Statični kadri

Vir: Sony DCR-TRV60E | format: DV | kontejner: AVI | ločljivost: 720×576 | FPS: 25 | bitna hitrost: 24.4 Mbps | trajanje: 1:00



Slika 21: Video posnetek dinamičnih kadrov na levi in statičnih na desni

Vir: Lasten

Ker je kamera snemala video po televizijskem sistemu, jaz pa želim video uporabiti na spletu, bom video pred stiskanjem tudi primerno obdelal in se s tem izognil morebitnim težavam. S pomočjo programa Adobe Premiere Pro, bom videu najprej odstranil prepletanje (angl. deinterlace), nato pa še spremenil razmerje pike na 1 : 1, kar bo pretvorilo ločljivost posnetka 720×576 , na ločljivost 1024×576 .

S stiskanjem želim dobiti video datoteke, ki so primerne tako za klasičen prenos preko interneta kakor za pretakanje, zato bom video stiskal po sledečih parametrih:

- Video visoke kakovosti: izvorna ločljivost (1024×576), podatkovna hitrost 5 Mbps
- Video srednje kakovosti: 640×360 , podatkovna hitrost 2,5 Mbps

- Video nizke kakovosti: 356×200 , podatkovna hitrost 600 Kbps

Tukaj nam video kodirniki nudijo celo paleto parametrov in izbiro različnih profilov. Preizkus vseh bi bil preobsežen, zato bom uporabljal splošne profile s privzetimi nastavitvami. Kodiranje bo potekalo v dveh ciklih kodiranja (angl. two pass encoding), s čimer želim dobiti boljše razmerje med kvaliteto zapisa in velikostjo datoteke. Video je prav tako mogoče stiskati z variabilno bitno hitrostjo, katere tukaj ne bom uporabil, zaradi omejitev nekaterih kodirnikov. (<http://vimeo.com/help/compression>, dne 16.9.2011)

Uporabljeni programi

Video posnetke sem pridobil s pomočjo programa *Adobe Premiere Pro*, ki je prav tako sposoben stiskanja videa, ne nudi pa podpore za vse formate, katere želim preizkusiti. Za stiskanje videa bo tukaj potrebno uporabiti tako imenovane video pretvornike (angl. video converters), ob katerih se lahko stvari precej zapletejo. Med njimi je izredno težko najti pretvornik, ki bi imel dobro podporo za vse formate, nudil vse potrebne nastavitve za kodiranje in hkrati deloval brezhibno. Tukaj lahko iz izkušenj rečem, da program, ki nudi vse v enem, ni dobra izbira, zato sem si pomagal z večjim številom programov hkrati. Programi, s katerimi sem lahko brez večjih težav opravil vse preizkuse so: *MediaCoder*, *VirtualDub* in *Microsoft Expression Encoder 4*. Vsi naštet programi so brezplačni, obstaja pa še veliko brezplačnih in plačljivih alternativ.

V preizkus sem želel prav tako vključiti format VP8, z uporabo kontejnerja WebM, vendar je ta povzročal preveč težav. Tega sem preizkušal z večjim številom programov, vendar je njegova uporaba pogosto povzročala napake v programih in med samim procesom stiskanja. Stiskanje je bilo dolgotrajno, končna datoteka pa ni bila stisnjena po parametrih, ki sem jih določil, kar ga je naredilo neprimernega za preizkus. S tem bo format VP8 izpuščen iz preizkusa.

Praktični primer 1

DATOTEKA: Oldtimerji – dinamični kadri.avi, velikost: 206 MB (216.065.082 B)			
UPORABLJEN FORMAT [KONTEJNER]	ČAS STIKSKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
H.264 [.mp4]	4:37	35,9 MB (37.659.465 B)	0,174 [83,6 %]
Theora [.ogg]	4:45	37,1 MB (38.911.210 B)	0,180 [82,0 %]
XviD [.avi]	3:48	33,5 MB (35.194.852 B)	0,163 [84,7 %]
WMV 9 [.wmv]	3:38	34,2 MB (35.933.444 B)	0,166 [83,4 %]
DATOTEKA: Oldtimerji – statični kadri.avi, velikost: 206 MB (216.062.160 B)			
H.264 [.mp4]	4:31	35,7 MB (37.503.494 B)	0,174 [82,6 %]
Theora [.ogg]	4:43	35,9 MB (37.746.819 B)	0,175 [82,5 %]
XviD [.avi]	3:32	32,3 MB (33.967.858 B)	0,157 [84,3 %]
WMV 9 [.wmv]	2:51	34,6 MB (36.325.444 B)	0,168 [83,2 %]

Tabela 14: Stiskanje videa visoke kakovosti: 1024×576 , bitna hitrost: 5 Mbps

Vir: Lasten

DATOTEKA: Oldtimerji – dinamični kadri.avi, velikost: 206 MB (216.065.082 B)			
UPORABLJEN FORMAT [KONTEJNER]	ČAS STIKSKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
H.264 [.mp4]	2:41	18,0 MB (18.885.323 B)	0,087 [91,3 %]
Theora [.ogg]	1:34	18,5 MB (19.441.501 B)	0,090 [91,0 %]
XviD [.avi]	3:17	16,9 MB (17.824.780 B)	0,082 [91,8 %]
WMV 9 [.wmv]	2:00	17,0 MB (17.909.444 B)	0,083 [91,7 %]
DATOTEKA: Oldtimerji – statični kadri.avi, velikost: 206 MB (216.062.160 B)			
H.264 [.mp4]	3:31	17,8 MB (18.735.756 B)	0,087 [91,3 %]
Theora [.ogg]	3:29	18,2 MB (19.149.430 B)	0,089 [91,1 %]
XviD [.avi]	3:19	17,7 MB (18.649.640 B)	0,086 [91,4 %]
WMV 9 [.wmv]	1:46	17,3 MB (18.197.444 B)	0,084 [91,6 %]

Tabela 15: Stiskanje videa srednje kakovosti: ločljivost: 640×360 , bitna hitrost: 2,5 Mbps

Vir: Lasten

DATOTEKA: Oldtimerji – dinamični kadri.avi, velikost: 206 MB (216.065.082 B)			
UPORABLJEN FORMAT [KONTEJNER]	ČAS STISKANJA	VELIKOST PO STISKANJU	RAZMERJE [PRIHRANEK]
H.264 [.mp4]	2:26	4,36 MB (4.581.020 B)	0,021 [97,9 %]
Theora [.ogg]	3:25	4,40 MB (4.620.236 B)	0,021 [97,9 %]
XviD [.avi]	3:14	3,98 MB (4.176.650 B)	0,019 [98,1 %]
WMV 9 [.wmv]	0:58	4,17 MB (4.377.944 B)	0,020 [98,0 %]
DATOTEKA: Oldtimerji – statični kadri.avi, velikost: 206 MB (216.062.160 B)			
H.264 [.mp4]	3:18	4,31 MB (4.524.326 B)	0,021 [97,9 %]
Theora [.ogg]	2:14	3,31 MB (3.477.716 B)	0,016 [98,4 %]
XviD [.avi]	3:25	4,33 MB (4.540.832 B)	0,021 [97,9 %]
WMV 9 [.wmv]	0:58	4,00 MB (4.197.944 B)	0,019 [98,1 %]

Tabela 16: Stiskanje videa nizke kakovosti: ločljivost: 356×200 , bitna hitrost: 600 Kbps

Vir: Lasten

9.8 Ugotovitve

Stiskanje videa je lahko boleče dolgotrajen in neprijeten proces. Tukaj gre predvsem za pretvarjanje med različnimi formati z določanjem novih parametrov. Programska oprema, ki izvaja tovrstno pretvarjanje mora imeti dobro podporo za prvotne formate videa, kot za novega, drugače se hitro srečamo s problemi, kot so omejitve parametrov, dolgotrajno stiskanje datoteke ali napake med stiskanjem. Neprimerno stiskanje lahko povzroči tudi napake pri predvajanju stisnjenega video zapisa. Dobra programska oprema je za učinkovito stiskanje videa tukaj bistvenega pomena, zato je toliko veliko govora tudi o kodirnikih, ki lahko s svojo kompleksnostjo in fleksibilnostjo, različno dobro obdelujejo video tokove.

S preizkusom stiskanja oziroma pretvarjanja video datotek, sem prišel do sledečih ugotovitev. Razlike med velikostmi stisnjenih datotek niso velike, njihova kvaliteta zapisa pa je večinoma dobra, konkurenca je. Na statično ali dinamično vsebino videa, se kodirniki odzivajo različno. Stiskanje statičnega videa načeloma traja dlje, saj ga je zaradi podobnosti, ki se ponavljajo med okvirji, mogoče bolj natančno stisniti. S tem so statični posnetki posledično, večinoma manjših velikosti, kakor dinamični. Razlike postajajo očitnejše tudi z višjim nivojem stiskanja. Z nižanjem ločljivosti in bitnih hitrosti, lahko privarčujemo na času, ki je potreben za stiskanje. Tukaj si kodirnik Xvid, za svoje delo vzame čas, ampak zna s tem privarčevati na velikosti datoteke. Xvid ima prav tako probleme s stiskanjem v VBR načinu in posnetkov nizkih ločljivosti, saj so ti po kvaliteti precej slabši od ostalih. Po času, ki je potreben za stiskanje, je najhitrejši med njimi bil WMV 9 (VC-1). Odprt video format Theora, se je

večinoma odrezal zelo solidno, medtem, ko je bilo pri Microsoftovem formatu WMV 9 opaziti, da varčuje z barvami bolj, kakor konkurenca. Na vseh področjih pa se je na splošno dobro izkazal H.264, kar je najverjetneje tudi razlog, za njegovo vse pogostejšo uporabo na spletu. Škoda, da kljub vloženemu trudu, v preizkus nisem mogel vključiti format VP8 in njegov kontejner WebM. Ta se med formati šteje za novinca in resnično se zdi, da še ni popolnoma dodelan in podprt. Delni preizkus, ki sem ga uspel opravil z njim, je namreč dal obetajoče rezultate.

Predlogi

Za učinkovito stiskanje in pretvarjanje videa, je potrebna dobra programska oprema. Tukaj odsvetujem uporabo programov, ki so poznani kot »vse v enem«, saj nam bodo ti večinoma nudili manj parametrov za stiskanje, prav tako pa so manj stabilni. Z uporabo dodatnih parametrov in profilov, je mogoče iz kodirnikov iztisniti še nekaj dodatnih izboljšav, ki lahko prispevajo k učinkovitejšem stiskanju. Bolj kot bitna hitrost podatkov pa na kvaliteto videa vpliva velikost ločljivosti, zato svetujem uporabo višjih ločljivosti, kakor bitnih hitrosti. Zaradi vedno večje uporabe videa na spletu, se tehnologije na tem področju prav tako razvijajo hitreje, zato je dobro biti na tekočem. Število video formatov je veliko, imajo svoje prednosti in slabosti, pri njihovi izbiri pa se je potrebno vprašati ali bomo ob njihovi uporabi dali prednost kvaliteti slike, hitrosti stiskanja ali velikosti datoteke.

10 ZAKLJUČEK IN UGOTOVITVE

Torej, kako učinkovito je stiskanje podatkov glede na to, koliko slabosti lahko nastane ob tem? Zelo učinkovito.

Kadar brskamo po spletu in prenašamo njegovo vsebino, se sploh ne zavedamo, da je skoraj vsa vsebina, ki jo uporabljamo, stisnjena. Niti se ne zavedamo, da se nam vsebina s katero delamo, avtomatsko stiska, kadar jo shranjujemo v datoteko. Ta tehnologija je namreč z nami že lep čas in je ob današnji poplavi informacij bolj potrebna, kot kadarkoli. Ob njeni uporabi se sicer lahko srečamo z vidno izgubo kvalitete digitalnega zapisa in drugimi nevšečnostmi s strani strojne in programske opreme, a vendar je to cena, katero smo pripravljeni plačati na račun dostopnosti, ki nam je s tem na razpolago.

Največji problem s katerim sem se srečal ob izdelavi te diplomske naloge je obsežnost, ki jo narekuje njen naslov. Področje stiskanja spletne vsebine je obsežno in zahtevno, literature s tega področja v Slovenskem jeziku pa malo. Nekateri formati so celo tako zaščiteni s strani razvijalcev, da je informacij o njih prav tako malo. Po drugi strani pa je tuja literatura s tega področja tako obsežna in kompleksna, da se lahko v njej hitro zgubiš. Ostali problemi s katerimi sem se srečeval, so bili bolj programske narave; še posebej na področju videa. S kompleksnostjo kompresijskih algoritmov, kodekov in formatov, rastejo tudi možnosti za morebitne težave, zato je potrebno biti pazljiv pri navajanju parametrov stiskanja in izbiri primerne programske opreme.

Sicer pa večjih problemov s samim procesom kompresiranja in dekompresiranja, ni bilo. Večina programov danes opravlja stiskanje že skoraj popolnoma avtomatsko. Če ob uporabi kvalitetnega izvora podatkov, namenimo tej tehnologiji vsaj malo pozornosti, bomo z rezultatom stiskanja najverjetneje presenečeni. Kadar se lotevamo kateregakoli tovrstnega stiskanja podatkov, močno priporočam, da se pred tem opravi tudi kratek preizkus. Tako se lahko najbolje prepričamo ali je stiskanje uspešno in ali nam rezultat stiskanja ustreza.

Svetovni splet in njegova vsebina iz dneva v dan brezmejno raste. Tukaj je tehnologija stiskanja podatkov samo ena izmed tovrstnih tehnologij, ki stojijo v ozadju svetovnega spleta in mu odpirajo nove poti. Kombinacije algoritmov in metod se še vedno razvijajo ter posledično prinašajo nove formate in nove programske zahteve. Te nam bodo najverjetneje

povzročale preglavice, prav tako pa nam bodo dajala nove možnosti, za še učinkovitejše stiskanje podatkov.

11 VIRI IN LITERATURA

11.1 Literatura

1. Austerberry, D.: *The Technology of Video and Audio Streaming, Second edition*, Focal Press, Anglija, 2005.
2. Salomon, D., Giovanni M.: *Handbook of Data Compression, Fifth edition*, Springer, Anglija, 2010.
3. Sayood, K.: *Introduction to Data Compression*, Academic Press, ZDA, 2000.

11.2 Spletni viri

1. <http://computer.howstuffworks.com/file-compression.htm>, dne 2.4.2011
2. <http://datacompression.info/>, dne 13.3.2011
3. <http://developer.yahoo.com/yui/2/>, dne 17.5.2011
4. <http://diveintohtml5.org/video.html>, dne 9.9.2011
5. <http://en.wikipedia.org/wiki/mkv>, dne 12.9.2011
6. <http://en.wikipedia.org/wiki/7z>, dne 15.5.2011
7. http://en.wikipedia.org/wiki/Advanced_Audio_Coding, dne 25.7.2011
8. http://en.wikipedia.org/wiki/Arithmetic_coding, dne 9.4.2011
9. http://en.wikipedia.org/wiki/Audio_data_compression#Lossless_compression, dne 1.9.2011
10. http://en.wikipedia.org/wiki/Chroma_subsampling, dne 23.8.2011
11. http://en.wikipedia.org/wiki/Comparison_of_audio_codecs, dne 30.7.2011
12. http://en.wikipedia.org/wiki/Comparison_of_container_formats, dne 8.9.2011
13. http://en.wikipedia.org/wiki/Comparison_of_video_codecs, dne 8.9.2011
14. http://en.wikipedia.org/wiki/Data_compression, dne 17.5.2011
15. http://en.wikipedia.org/wiki/Data_compression, dne 20.3.2011
16. <http://en.wikipedia.org/wiki/DEFLATE>, dne 10.4.2011
17. <http://en.wikipedia.org/wiki/Digitizing>, dne 5.4.2011
18. <http://en.wikipedia.org/wiki/File:Colorcomp.jpg>, dne 2.9.2011
19. http://en.wikipedia.org/wiki/File:JPEG_example_JPG_RIP_001.jpg, dne 21.8.2011
20. http://en.wikipedia.org/wiki/File:JPEG_example_JPG_RIP_010.jpg, dne 21.8.2011

21. http://en.wikipedia.org/wiki/File:JPEG_example_JPG_RIP_025.jpg, dne 21.8.2011
22. <http://en.wikipedia.org/wiki/File:VectorBitmapExample.svg>, dne 24.8.2011
23. <http://en.wikipedia.org/wiki/Flac>, dne 26.7.2011
24. <http://en.wikipedia.org/wiki/H.264>, dne 11.9.2011
25. http://en.wikipedia.org/wiki/Huffman_coding, dne 10.4.2011
26. http://en.wikipedia.org/wiki/Image_compression, dne 30.7.2011
27. <http://en.wikipedia.org/wiki/Jpeg>, dne 23.8.2011
28. http://en.wikipedia.org/wiki/List_of_common_resolutions, dne 5.9.2011
29. http://en.wikipedia.org/wiki/LZ77_and_LZ78, dne 10.4.2011
30. <http://en.wikipedia.org/wiki/MP3>, dne 25.7.2011
31. <http://en.wikipedia.org/wiki/MPEG-4>, dne 10.9.2011
32. http://en.wikipedia.org/wiki/MPEG-4_Part_14, 12.9.2011
33. <http://en.wikipedia.org/wiki/Ogg>, dne 12.9.2011
34. http://en.wikipedia.org/wiki/Prediction_by_partial_matching, dne 9.4.2011
35. http://en.wikipedia.org/wiki/Quantization_%28signal_processing%29, dne 9.4.2011
36. <http://en.wikipedia.org/wiki/RAR>, dne 15.5.2011
37. http://en.wikipedia.org/wiki/Run-length_encoding, dne 9.4.2011
38. http://en.wikipedia.org/wiki/Speech_coding, dne 2.9.2011
39. http://en.wikipedia.org/wiki/Streaming_media, dne 24.3.2011
40. <http://en.wikipedia.org/wiki/SVG>, dne 23.8.2011
41. <http://en.wikipedia.org/wiki/Theora>, dne 11.9.2011
42. http://en.wikipedia.org/wiki/Video_compression, dne 8.8.2011
43. <http://en.wikipedia.org/wiki/Vorbis>, dne 26.7.2011
44. <http://en.wikipedia.org/wiki/VP8>, dne 11.9.2011
45. <http://en.wikipedia.org/wiki/Webm>, dne 12.9.2011
46. http://en.wikipedia.org/wiki/Windows_Media_Audio, dne 26.7.2011
47. http://en.wikipedia.org/wiki/Windows_Media_Video, dne 11.9.2011
48. <http://en.wikipedia.org/wiki/Xvid>, dne 11.9.2011
49. http://en.wikipedia.org/wiki/ZIP_%28file_format%29, dne 15.5.2011
50. http://portableapps.com/apps/music_video/vlc_portable, dne 17.5.2011
51. http://vector-conversions.com/images/raster_vs_vector_1.jpg, dne 2.8.2011
52. <http://vimeo.com/help/compression>, dne 16.9.2011

53. <http://www.data-compression.com/index.shtml>, dne 14.3.2011
54. <http://www.dcs.warwick.ac.uk/research/mcg/bmmc/index.html>, dne 16.9.2011
55. <http://www.dcs.warwick.ac.uk/research/mcg/bmmc/index.html>, dne 16.9.2011
56. <http://www.edrawsoft.com/images/cliparts/clipart-car.png>, dne 3.8.2011
57. http://www.fileformat.info/mirror/egff/ch09_01.htm, dne 17.6.2011
58. <http://www.fileinfo.com/extension/docx>, dne 20.5.2011
59. <http://www.http-compression.com/>, dne 9.5.2011
60. <http://www.mediacollege.com/audio/images/loudspeaker-waveform.gif>, dne 29.8.2011
61. <http://www.microsoft.com/windows/windowsmedia/howto/articles/vc1techoverview.aspx>, dne 9.9.2011
62. <http://www.openclipart.org/detail/131845/water-melon-by-gnokii>, dne 24.8.2011
63. http://www.planetoftunes.com/digiaudio/how_a_to_d.html, dne 28.8.2011
64. <http://www.reelseo.com/file-formats-containers-compression/>, dne 17.3.2011
65. <http://www.streaminglearningcenter.com/articles/streaming-101-the-basics---codecs-bandwidth-data-rate-and-resolution.html>, dne 30.8.2011
66. http://www.w3schools.com/xml/plant_catalog.xml, dne 17.5.2011
67. <http://www.websiteoptimization.com/speed/tweak/compress/>, dne 3.5.2011
68. <http://www.websiteoptimization.com/speed/tweak/compress/>, dne 9.5.2011