Między innymi o :

- *Warunki KODA*
- Co to jest kompresja?
- Rola kompresji
- Dane
- Stratność procesu kompresji
- Efektywność
- Historia
- Proste przykłady kodowania

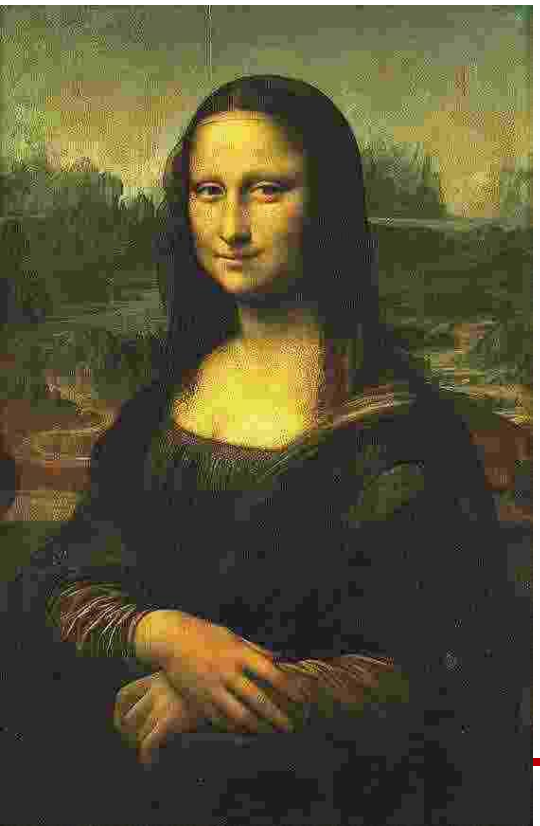
WPROWADZENIE

CZYM JEST KOMPRESJA?

Co to jest kompresja?

Def. 1 (węższa) Kompresja to odwracalny lub nieodwracalny proces redukcji bitowego rozmiaru reprezentacji danych źródłowych

Def. 2 (szersza) Kompresja to wyznaczanie możliwie użytecznej w określonym zastosowaniu reprezentacji danych źródłowych (czyli wyznaczanie reprezentacji informacji)



Mona Lisa

http://en.wikipedia.org/wiki/Image:Mona_Lisa.jpg

Jak kompresować?

Lenna, 512x512, 8bpp

Lenna, 128x128

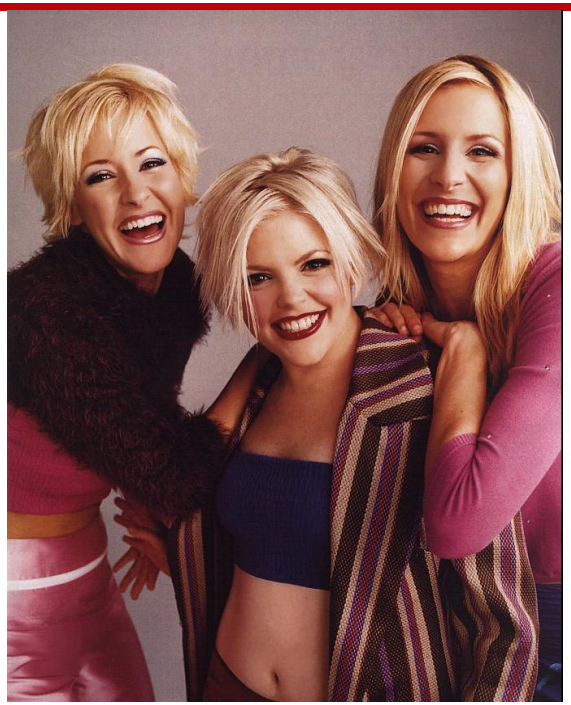
Lenna, 512x512, 3bpp

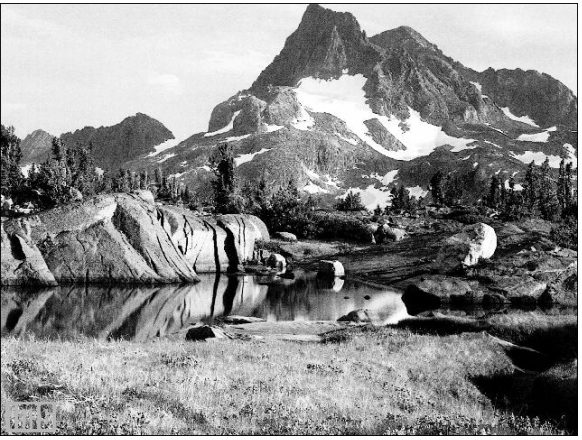
Lenna, 512x512, 0.5 bpp (JPEG)

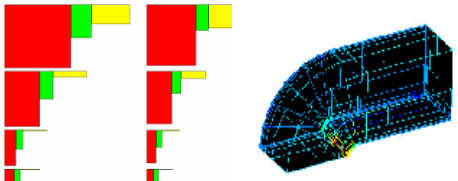
Dane

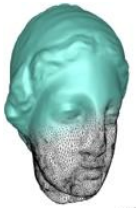
- Teksty
- Obrazy
- Dźwięk
- Dane pomiarowe
- Dokumenty (dane mieszane)
- Katalogi, dyski
- …

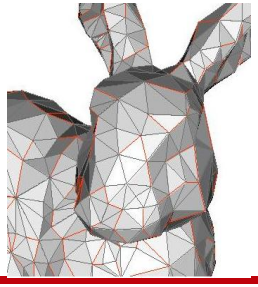
Dane: obrazy 1/2

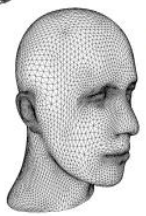




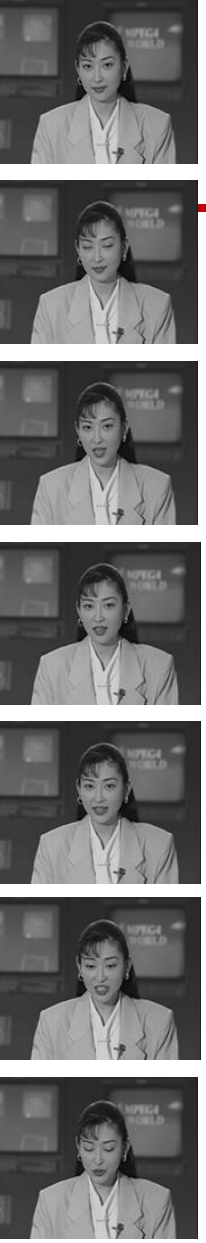


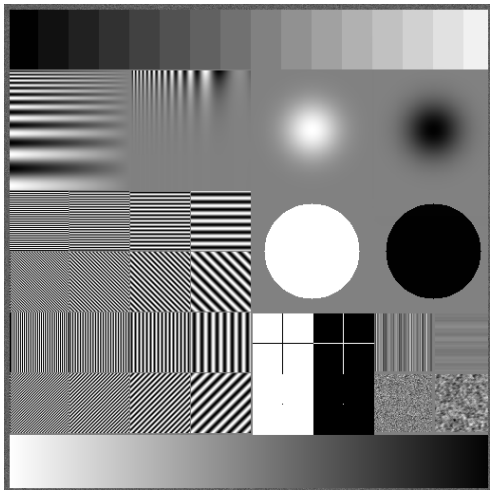


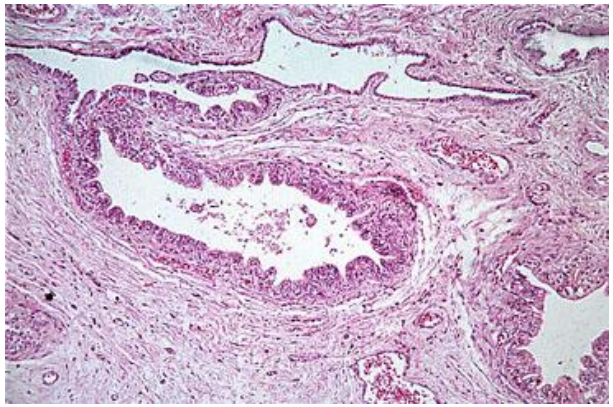


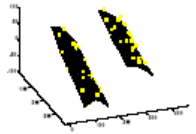


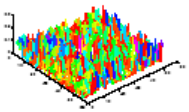
Dane: obrazy 2/2











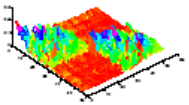


Figure 14.6. Motion segmentation – experimental results: top-left: one frame from a sequence of pictures of two cylinders, including feature tracks; top-right: the recovered shapes after motion segmentation; bottom-left: the shape interaction matrix; bottom-right: the matrix after sorting. Reprinted from [Costeira and Kanade, 1998], Figures 13–15.

14.7 Assignments

Exercises

1. In this exercise we prove Theorem 4. Let us define

$$a(\mathbf{v}) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{v}\cdot\mathbf{a}_i)^2,$$

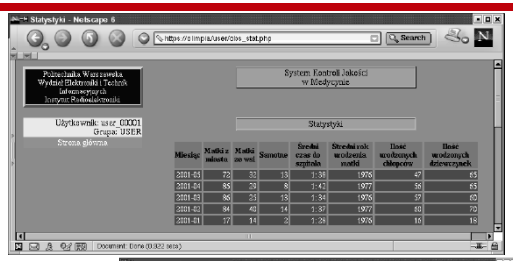
$\mathcal{S} = (\mathbf{a}_1, \ldots, \mathbf{a}_n)$, and $\mathcal{C} = \frac{1}{n}\mathcal{S}\mathcal{S}^T$. With this notation we have

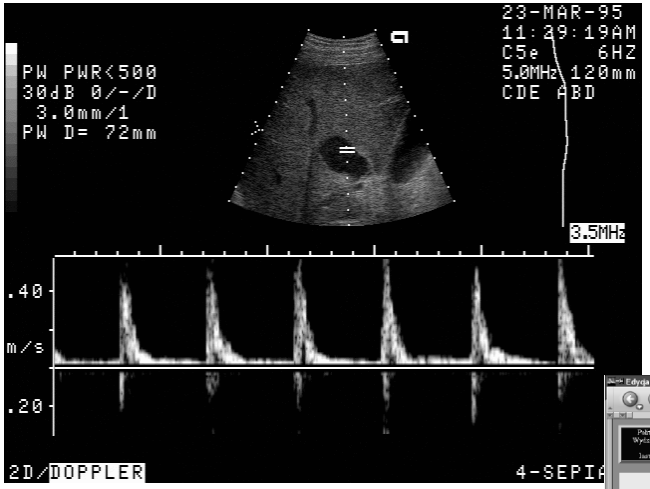
$$a(\mathbf{v}) = \mathbf{v}^T\mathcal{C}\mathbf{v}.$$

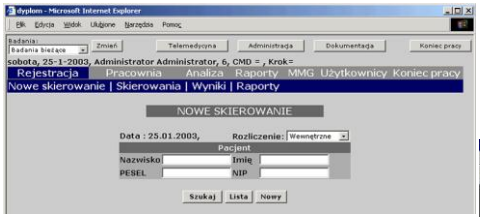
You can assume for simplicity that the eigenvalues of $\mathcal{C}$ are all distinct. Use the following steps to prove Theorem 4.

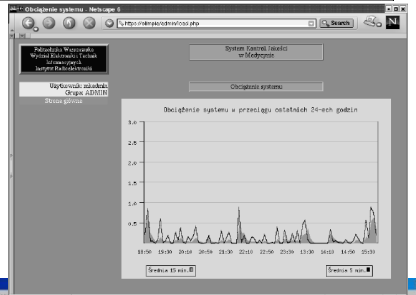
(a) Show that determining V_p reduces to constructing the orthonormal family of vectors $\mathbf{v}_i$ $(i = 1, \ldots, p)$ that maximizes $A \stackrel{\text{def}}{=} \sum_{i=1}^{p} a(\mathbf{v}_i)$.

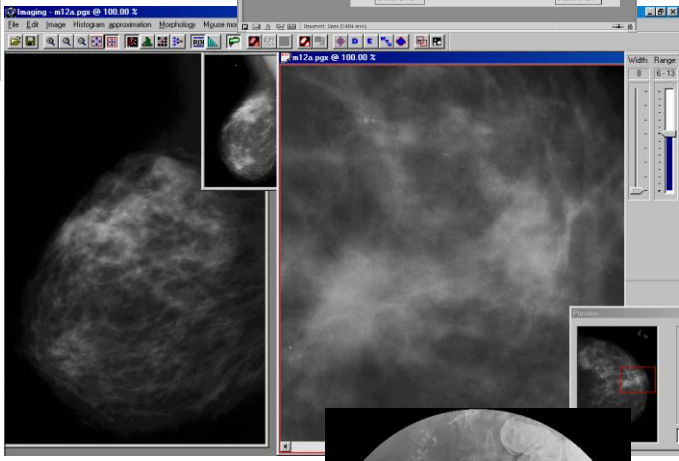
Dane: zastosowania medyczne

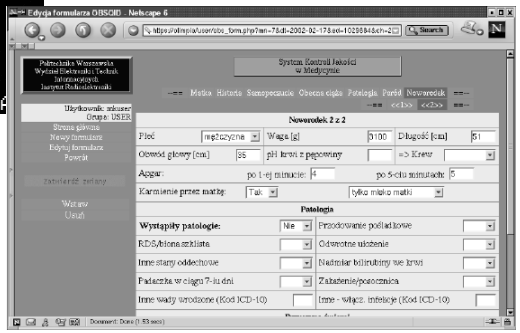


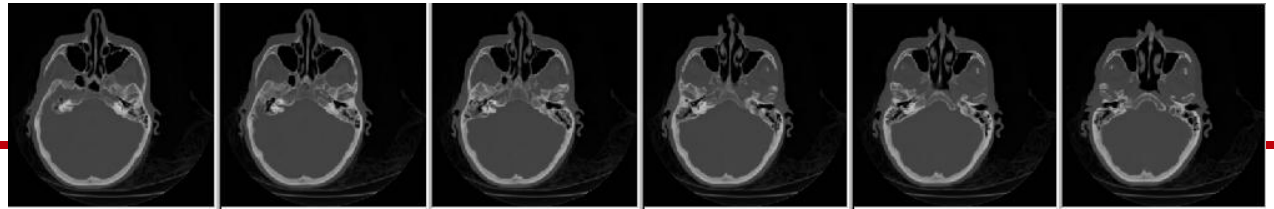












Uniwersalne zestawy danych testowych

- Calgary Corpus

http://links.uwaterloo.ca/calgary.corpus.html

Name	Size	Description
bib	111,261	Bibliographic files (in the Unix "refer" format)
book1	768,771	A book "Far from the Madding Crowd" by Thomas Hardy
book2	610,856	A book "Principles of Computer Speech" by Ian Witten
geo	102,400	Geophysical data of seismic activity
news	377,109	Postings from various newsgroups on USENET
obj1	21,504	VAX executable of program "progp"
obj2	246,814	Macintosh executable of "Knowledge support system"
paper1	53,161	A paper "Arithmetic coding for data compression" by Ian Witten, Radford Neal, and John Cleary
paper2	82,199	A paper "Computer (in)security" by Ian Witten
paper3	46,526	A paper "In search of autonomy" by Ian Witten
paper4	13,286	A paper "Programming by example revisited" by John Cleary
paper5	11,954	A paper "A logical implementation of arithmetic" by John Cleary
paper6	38,105	A paper "Compact hash tables using bidirectional linear probing" by John Cleary
pic	513,216	Picture number 5 from the CCITT Facsimile test files (text + drawings)
progc	39,611	C source code of Unix compress version 4.0
progl	71,646	LISP source code
progp	49,379	Pascal source code of Prediction by Partial Matching evaluation program
trans	93,695	Transcript of a session on a EMACS terminal

Uniwersalne zestawy danych testowych

- Cantenbury corpus

  http://corpus.canterbury.ac.nz/

Name	Size	Description
alice29.txt	152,089	A book "Alice's Adventures in Wonderland" by Lewis Carroll
asyoulik.txt	125,179	A play "As you like it" by William Shakespeare
bible.txt	4,047,392	The King James version of the Bible
cp.html	24,603	Compression pointers
E.coli	4,638,690	Complete genome of the Esxherichia coli bacterium
fileds.c	11,150	C source code
grammar.lsp	3,721	LISP source code
kennedy.xls	1,029,774	Excel spreadsheet
lcet10.txt	426,754	Proceedings from "Workshop on electronic texts"
plrabn12.txt	481,861	A book "Paradise Lost" by John Milton
ptt5	513,216	Picture number 5 from the CCITT Facsimile test files (text + drawings)
sum	38,240	SPARC executable
world192.txt	2,473,400	The CIA world factbook
xargs.1	4,227	GNU manual page of xargs

Uniwersalne zestawy danych testowych

- Silesia Corpus

http://www.data-compression.info/Corpora/SilesiaCorpus/

Filename	Description	Type	Source	Raw size [B]
dickens	Collected works of Charles Dickens	English text	Project Gutenberg	10,192,446
mozilla	Tarred executables of Mozilla 1.0 (Tru64 UNIX edition)	exe	Mozilla Project	51,220,480
mr	Medical magnetic resonanse image	picture	Hospital image	9,970,564
nci	Chemical database of structures	database	CACTVS Chemical Information Services at LMC/NCI	33,553,445
ooffice	A dll from Open Office.org 1.01	exe	Open Office	6,152,192
osdb	Sample database in MySQL format from Open Source Database Benchmark	database	Open Source Database Benchmark Project	10,085,684
reymont	Text of the book Chłopi by Władysław Reymont	Polish pdf	Virtual Library of Polish Literature	6,627,202
samba	Tarred source code of Samba 2-2.3	src	Samba Project	21,606,400
sao	The SAO star catalog	bin data	Astronomica Catalogs and Catalog Formats	7,251,944
webster	The 1913 Webster Unabridged Dictionary	html	Project Gutenberg	41,458,703
xml	Collected XML files	html	XMLPPM: XML-Conscious PPM Compressio	5,345,280
x-ray	X-ray medical picture		Hospital image	8,474,240
Total				211,938,580

WARUNKI REALIZACJI PRZEDMIOTU

Przedmiot

- Celem poznanie w zakresie
  - podstaw teoretycznych (teoria informacji, teoria zniekształceń źródeł inf.)
  - aktualnych paradygmatów kompresji
  - algorytmów kodowania danych
  - zasad realizacji prostych algorytmów kompresji
  - przeglądu współczesnych narzędzi i standardów (zastosowania)
  - doboru koderów optymalnych (analiza, kryteria)
- Umiejętności
  - posługiwanie się syntetyczną i pragmatyczną wiedzą w zakresie nowoczesnych i użytecznych metod kompresji danych
  - konstrukcja efektywnych algorytmów kompresji
  - optymalizacja kodeków bazujących na otwartych bibliotekach według kryteriów dopasowanych do charakteru zastosowań
  - eksperymentalna weryfikacja, ocena użyteczności i dobór kodeków
- Ocena
  - egzamin 30 pkt
  - projekt 30 pkt: plan 5, realizacja 15, weryfikacja-sprawozdanie 10
  - dodatkowe punkty za aktywność (max 10)
  - projekt zaliczamy do końca semestru

Projekt

- Istotne umiejętności rozwiązania problemu praktycznego
- Algorytmika, rzadziej rozważania teoretyczne, jeszcze rzadziej dokonania programistyczne
- Cechy ogólne:
  - C/C++
  - prosty interfejs obsługi
  - rozbudowany interfejs prezentacji wyników
  - rozdzielenie kodera i dekodera
  - wyczerpujące sprawozdanie wkładu własnego (algorytm, oprogramowanie, eksperymenty)
- 4 osoby w zespole
- Model pracy systematycznej: plan (połowa listopada), realizacja (koniec grudnia), sprawozdanie - dodatkowe profity za projekty wczesne (oddane przynajmniej tydzień przed końcem semestru)

Warunki: literatura

- **A. Przelaskowski, „Kompresja danych: podstawy, metody bezstratne, kodery obrazów”, BTC, 2005**
- **Materiały na stronie: http://www.ire.pw.edu.pl/~arturp/**
  ***wybierz* Dydaktyka, *potem* KODA**
- K. Sayood, „Introduction to Data Compression”, Third Edition, Morgan Kaufmann Publishers, 2006 (wyd. pol: „Kompresja danych: wprowadzenie”, READ ME, 2002)
- D. Salomon, „A concise introduction to data compression”, Springer, 2008
- M. Nelson, „The Data Compression Book”, 2nd edition, MIS:Press,1995
- W. Skarbek, „Metody reprezentacji obrazów cyfrowych”, Akademicka Oficyna Wydawnicza PLJ, W-wa 1993
- W. Skarbek, „Multimedia. Algorytmy i standardy kompresji”, Akademicka Oficyna Wydawnicza PLJ, W-wa 1998
- A. Drozdek, „Wprowadzenie do kompresji danych”, WNT, 1999
- M. Rabbani, P. W. Jones, „Digital Image Compression Techniques”, SPIE Press, 1991
- M. Domański, „Zaawansowane techniki kompresji obrazów i sekwencji wizyjnych”, Wydawnictwo Politechniki Poznańskiej, 2000

PODSTAWOWE ZAGADNIENIA – WSTĘPNY ZARYS PROBLEMU

Rola kompresji

- Warunkująca rozwój technologiczny: reprezentacje danych źródłowych
- Filozoficzna: *naczynia i nerwy* jako symbol komunikacji warunkującej życie
- Społeczna w kontekście organizacji życia:
  - era informacji/wiedzy(?)
  - społeczeństwo sieciowe (Castells)
  - cena chwili
  - trzecia kultura integracji (w uproszczeniu: *integracja człowiek- komputer*)
- Użytkowa w kontekście wymiany i gromadzenia informacji: rejestracja, przetwarzanie, archiwum, transmisja, prezentacja (odbiór)

Podstawowe pojęcia: odwracalność

- Odwracalność numeryczna

- Odwracalność percepcyjna (psychowizualna)
- Odwracalność lokalna
- Odwracalność semantyczna
- Odwracalność syntaktyczna

- Selekcja informacji

Podstawowe pojęcia: efektywność

- Stopień kompresji (procent, średnia bitowa)
- Czas
- Jakość
- Iloczyn stopnia i jakości
- Uporządkowanie (hierarchia)
- Odporność na zakłócenia
- Implementacja (zrównoleglenie, oszczędność pamięci itd.)
- Adaptacyjność
- Elastyczność
- Uniwersalność
- Skalowalność
- Kontrola długości reprezentacji kodowej
- …

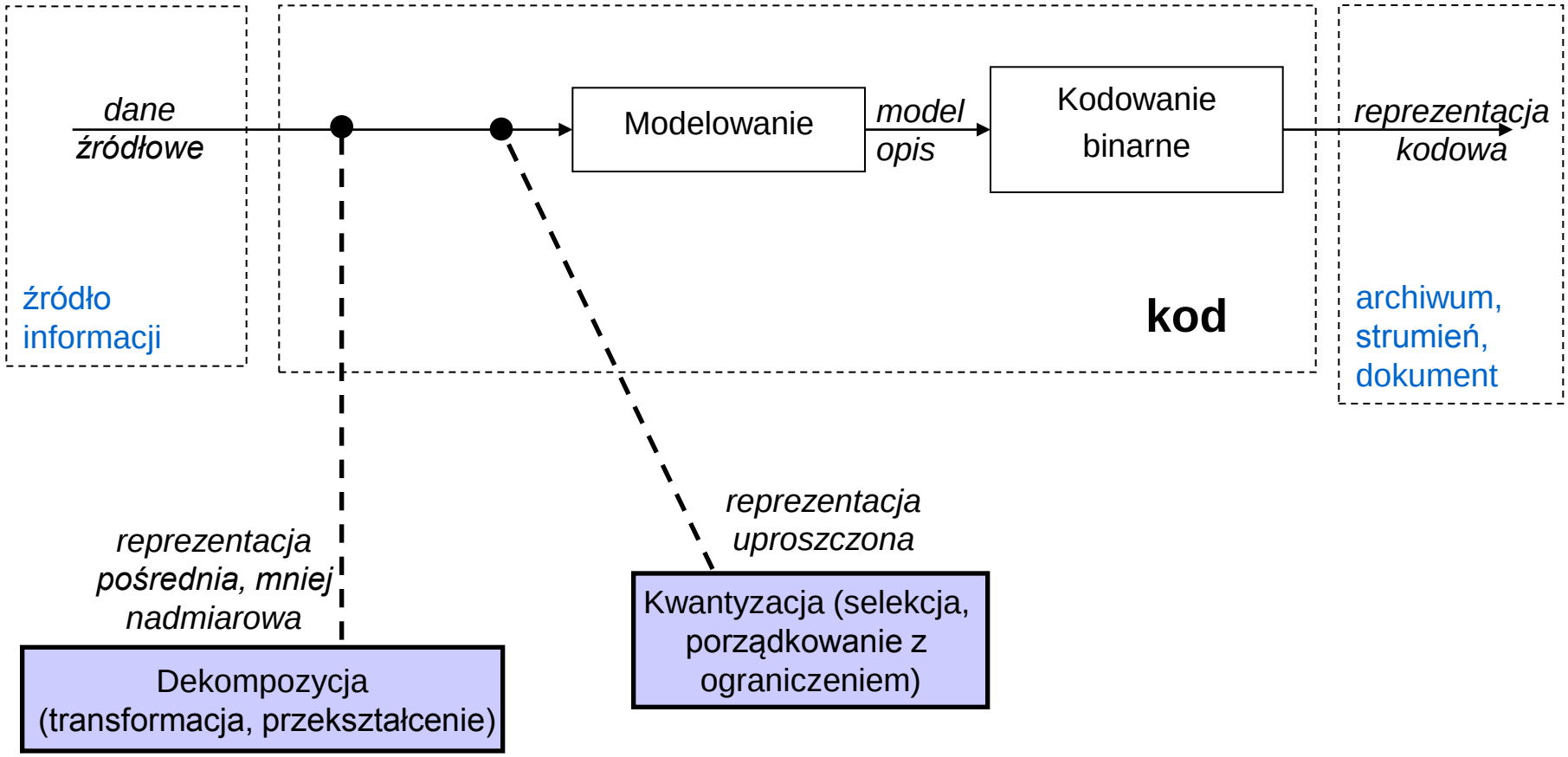
Historia

- Sygnały dymne, czyli pisanie krótkich listów: potrzeba czasu, by wyszło krócej (B.Pascal: *mój list jest dłuższy niż zwykle, gdyż nie miałem czasu, żeby napisać krócej*)
- Shannon, przełom lat 40-50, podstawy statystycznej teorii informacji
- Kod Huffmana, 1952
- Metody stratne (ekstrakcyjne), lata 60
- Metody słownikowe, lata 70
- Kod arytmetyczny i transformacyjne kodowanie, lata 80
- Standardy JPEG, MPEG, początek lat 90
- Metody falkowe, lata 90
- JPEG2000, rozszerzenia MPEG-4, czyli szerokie rozumienie kompresji, początek XXI wieku
- Archiwizery, formaty dokumentów, nowe transformacje, selektory informacji ….

Podstawowe pojęcia w kompresji danych

- KOD – reguła (lub algorytm) tworzenia reprezentacji kodowej (efektywnego ciągu bitowego) danych źródłowych
- Kodek – realizacja algorytmu kodowania (oprogramowanie, sprzęt)
- Kodowanie – proces tworzenia reprezentacji kodowej (koderem według ustalonego kodu)

Kodowanie (paradygmat podstawowy)



Modelowanie (1 etap kodowania źródłowego)

- pomysł, inteligencja, mózg
- stworzenie modelu probabilistycznego, deterministycznego, dziedzinowego (przekształceń), przybliżonego, mieszanego
- dobór charakteru modelu: adaptacyjnego, statycznego, z nauczaniem, z przełączaniem, ze słownikiem itp.
- wykorzystanie przekształcenia do reprezentacji pośredniej (potencjalnie nienadmiarowej, uporządkowanej, wyselekcjonowanej)

Kody binarne (2 etap kodowania źródłowego)

- silnik, napęd
  - konsekwencja **modelu**
  - reguła minimalizacji wyjściowego ciągu bitów
- wykorzystanie konkatenacji **słów kodowych**
  - słowa przypisane symbolom, blokom symboli, stanom modelu
  - kodowanie przyrostowe
- jednoznaczna dekodowalność
  - różnicowanie długości słów (lub ich części)
  - unikanie nadmiarowości

PROSTE REALIZACJE KODÓW

Przykłady

Kod dwójkowy (dane źródłowe):

- alfabet źródła informacji: $A_S=\{a_0,a_1,\ldots,a_{n-1}\}$
- reguła: $B_k(a_i)= \xi_i=(i)_{2,k}$
- parametr $k=\lceil \log_2 n \rceil$, gdzie n to liczba możliwych postaci danych źródłowych

Np.

A_S={‘ola’,’jola’,’kasia’,’basia’},

wtedy n=4, k=2 oraz

alfabet słów kodowych A_{B2}={00,01,10,11}

Zakodujmy!

We: $\mathbf{s}_{we}$ = (5, 5, 5, 2, 2, 11, 11, 11, 11, 11, 8)

$A_S = \{0,1,2,\ldots,15\}$

$B_4(\mathbf{s}_{we})$= 0101 0101 0101 0010 0010 1011 1011 1011 1011 1011 1000 - długość: 44 bity

M: $P(\mathbf{s}_{we}) = ((3, 5), (2, 2), (5, 11), (1, 8)) = ((l_i, s_i))_{i=1,2,\ldots}$

Wy1: $B_3(l_i\text{-}1)B_4(s_i)_{i=1,2,\ldots.}$=0100101 0010010 1001011 0001000 - długość 28 bitów

M': wagi symboli (kolejno 3(5), 2(2), 5(11), 1(8)

Wy2: $K_{VLC}(5)$=10, $K_{VLC}(2)$=110, $K_{VLC}(11)$=0, $K_{VLC}(8)$=111

$K_{VLC}(\mathbf{s}_{we})$=10 10 10 110 110 0 0 0 0 0 111 - 20 bitów + nagłówek

M': $P''(\mathbf{s}_{we}) = \{r_i : r_i = s_i - s_{i-1},\ i = 1, \ldots, 11,\ s_0 = 0\}=$

$=\{5, 0, 0, -3, 0, 9, 0, 0, 0, 0, -3\}$

M'': wagi symboli (kolejno 1(5), 7(0), 2(-3), 1(9)

Wy3: $K_{VLC}(5)$=110, $K_{VLC}(0)$=0, $K_{VLC}(-3)$=10, $K_{VLC}(9)$=111

$K_{VLC}(\mathbf{s}_{we})$=110 0 0 10 0 111 0 0 0 0 10 - 17 bitów + nagłówek

Przykład: RLE w PCX

Kodujemy kolejne serie symboli (l_k, s_k) według zasady:

- jeśli długość serii $l_k=1$ i symbol $s_k=a_i<192$

  $\xi_j = B_8(a_i)$, czyli tylko kod dwójkowy
- w przeciwnym wypadku

  $\xi_j = K_{powt}(l)\, B_8(a_i)$, $l_k = l < 64$

  $A_{Kpowt} = \{11B_6(l) : l = 1, \ldots, 63\}$

Przykład:

- $\xi_1 = K_{powt}(10)K_{symb}(1) = 11001010\ 00000001$
- $\xi_2 = K_{powt}(64)K_{symb}(254) = K_{powt}(63)K_{symb}(254)\ K_{powt}(1)K_{symb}(254) =$
  $= 11111111\ 11111110\ 11000001\ 11111110$
- $\xi_3 = K_{powt}(1)K_{symb}(5) = 00000101$
- $\xi_4 = K_{powt}(1)K_{symb}(201) = 11000001\ 11001001$

RLE 2W (modyfikacja RLE z PCX)

a)

5	5	17	17	17
5	5	10	17	10
0	0	0	0	0
0	0	0	0	5
8	8	8	5	5

b)

85	90	100	100	100
85	100	100	100	100
85	85	100	100	100
85	120	85	100	100
85	120	85	100	100

Zasada sąsiedztwa: 00 - z góry, 01 - góra-lewo skos, 10 - góra-prawo skos, a 11 - w wierszu
Kodowanie: sąsiedztwo + 6 bitów długości serii (+ symbol jedynie dla 11)

Etap	Sekwencja danych
Modelowanie	(2w, 5), (3w, 17), (2g), (1w, 10), (1g), (1w, 10), (5w, 0), (4g), (1w, 5), (3w, 8), (2w, 5)
Binarne kodowanie	C205 C311 02 C10A 01 C10A C500 04 C105 C308 C205
Modelowanie	(1w, 85), (1w, 90), (3w, 100), (1g), (4w, 100), (2w, 85), (3g), (1g), (1w, 120), (3gl), (5g)
Binarne kodowanie	C155 C15A C364 01 C464 C255 03 01 C178 43 05

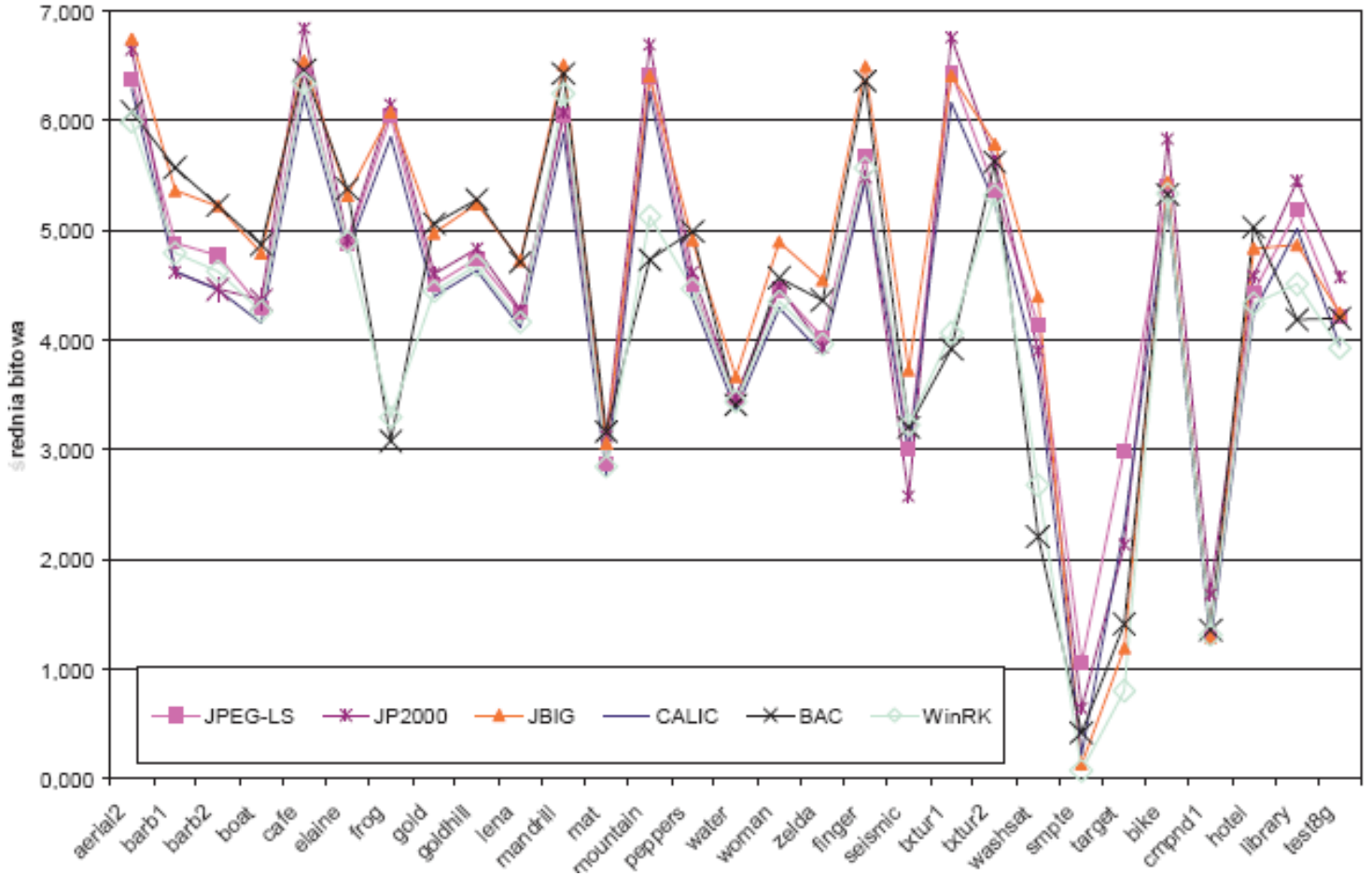
Przykładowe wyniki testów

		barb1.pgm	boat.pgm	goldhill2.pgm	lena.pgm	mandrill.pgm	zelda.pgm	średnia
RLE	Zygzakowy	7,1	6,15	6,65	5,96	7,8	5,76	***6,57***
	Naprzemienny	6,88	5,89	6,19	5,63	7,51	5,44	***6,26***
	Hilbert	6,87	5,81	6,26	5,73	7,55	5,41	***6,27***
	Po wierszach	7,07	5,88	6,2	5,88	7,38	5,69	***6,35***
RLE 2D v1	Zygzakowy	6,9	6,01	6,22	5,74	7,29	5,4	***6,26***
	Naprzemienny	6,69	5,76	5,81	5,42	7,01	5,13	***5,97***
	Hilbert	6,72	5,79	5,95	5,6	7,08	5,16	***6,05***
	Po wierszach	7,13	6,07	6,2	6,04	6,98	5,73	***6,36***
RLE 2D v2	Zygzakowy	6,81	5,93	6,1	5,64	7,22	5,26	***6,16***
	Naprzemienny	6,6	5,68	5,68	5,32	6,93	4,99	***5,87***
	Hilbert	6,64	5,7	5,83	5,5	7,01	5,02	***5,95***
	Po wierszach	7,07	6	6,11	5,9	6,9	5,62	***6,27***
PCX		8,70	8,20	8,47	8,49	8,34	7,96	*8,36*
PCX + AC		6,53	5,77	5,89	5,57	7,11	5,75	***6,10***

w tabeli wartości średnich bitowych

Paradygmaty dziś (1/2)

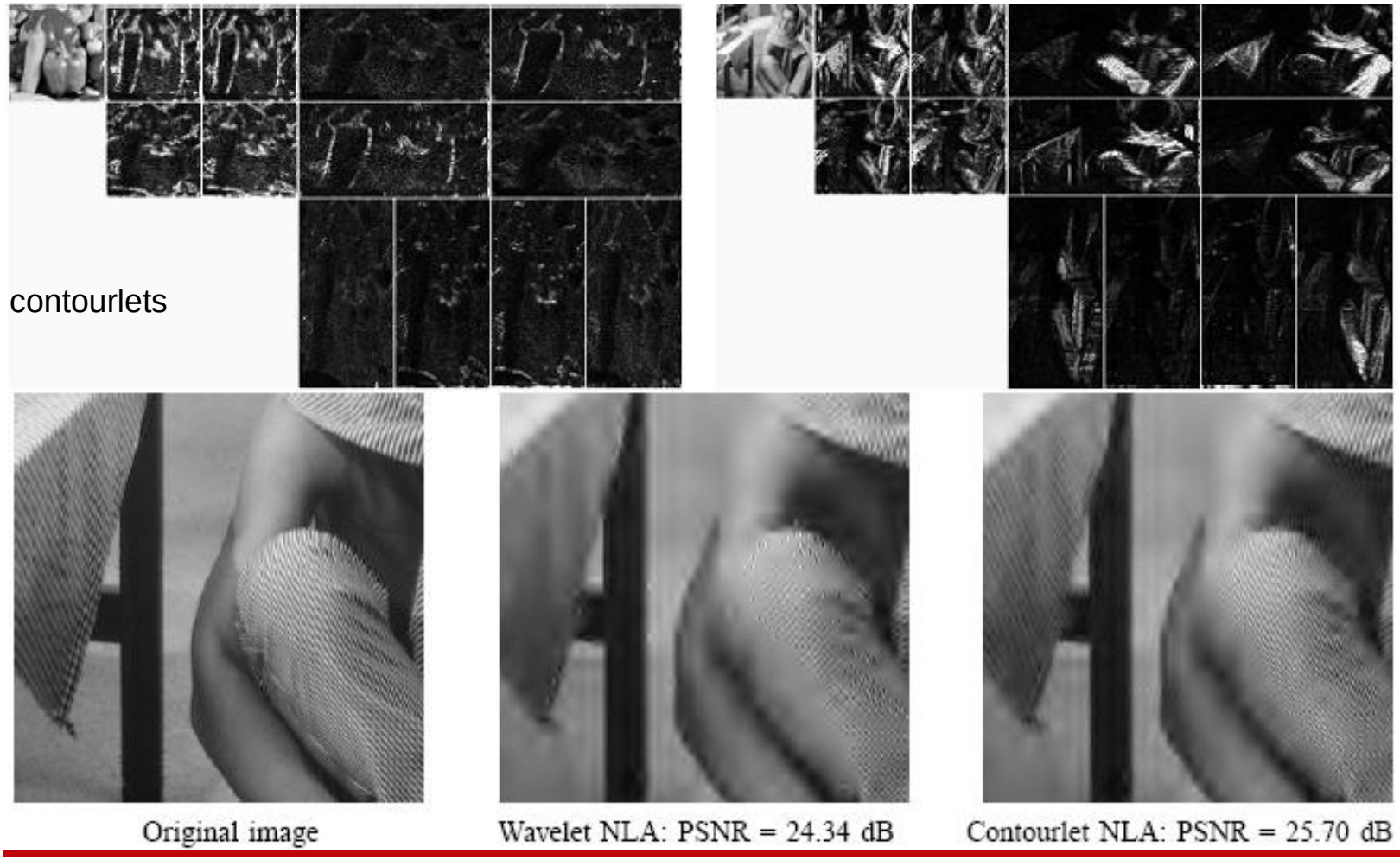
- Archiwizery (uniwersalne narzędzia do archiwizacji danych)



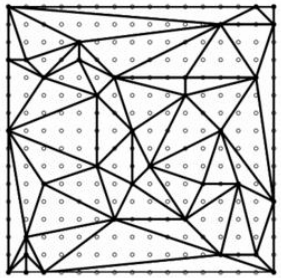
4,55 bpp (JPEG-LS), 4,60 bpp (JPEG2000), 4,75 bpp (JBIG), 4,35 bpp (CALIC), 4,36 (BAC) oraz 4,1 bpp (WinRK v. 2.1.6)

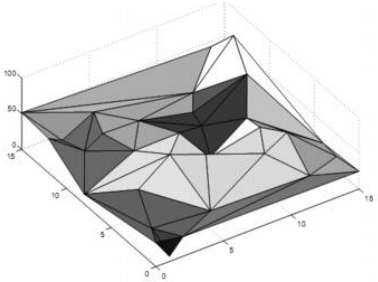
Paradygmaty dziś (2/2)

- Specjalizowane narzędzie kompresji: elastyczny koder skalowalny (kompresja selektywna)

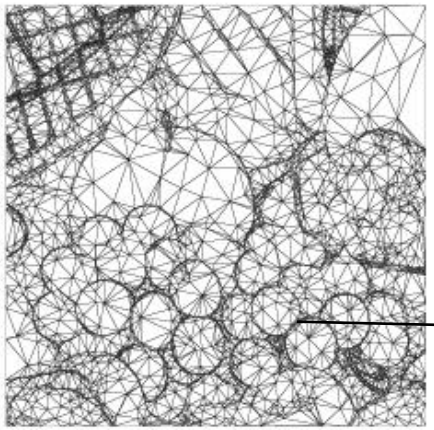


Przykłady elastycznej, selektywnej kompresji – różne modele obrazów

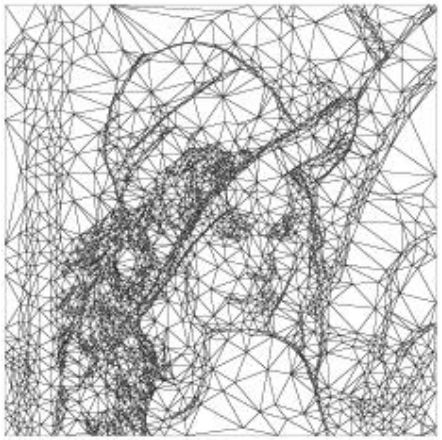


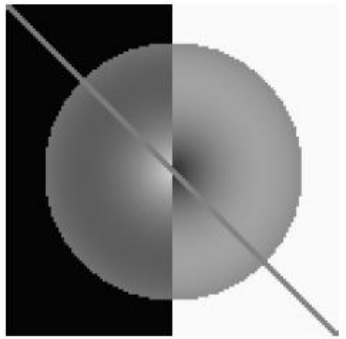


AT*: 0.18 bpp, 32.38 dB

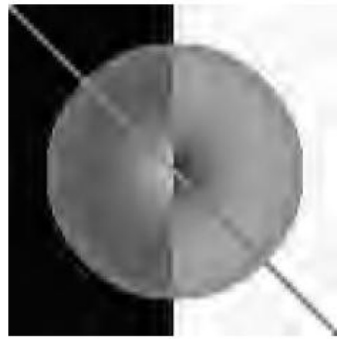


AT*: 0.15 bpp, 31.48 dB

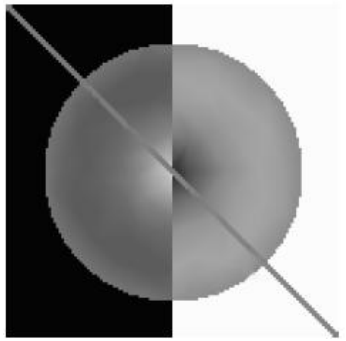


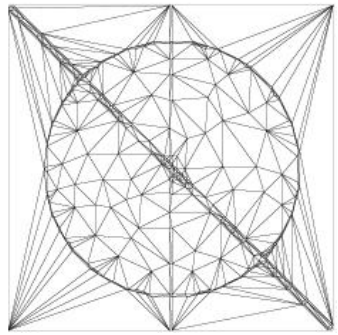


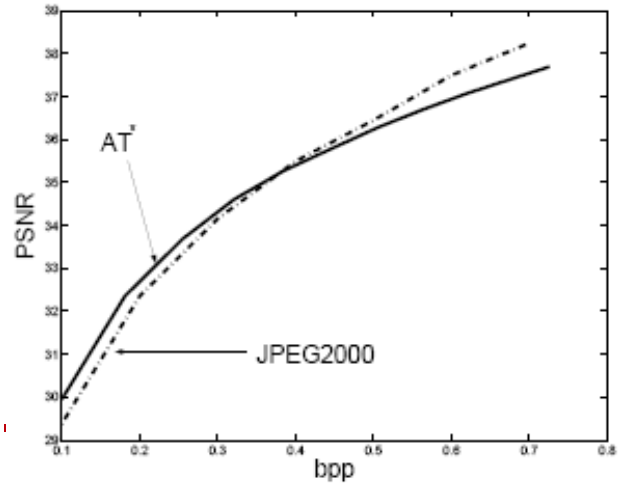
Reflex: 128 × 128



JPEG2000: 0.251 bpp, 28.74 dB







L. Demaret, N. Dynb, A. Iske (2006) Image compression by linear splines over adaptive triangulations. Signal Processing 86:1604-–16.