Compresie near-lossless (cu pierderi controlate)

I. Câte ceva despre distanțe d.p.d.v. matematic

În matematică o funcție distanță pe un set dat M este o funcție d: $M \times M \rightarrow \mathbf{R}$, unde **R** reprezintă un set de numere reale, care satisface următoarele condiții:

1. Nenegativitatea

$d(x,y) \geq 0$, şi $d(x,y) = 0$ dacă şi numai dacă $x = y$.

Distanța este nenegativă între orice două puncte iar dacă distanța este 0 punctele sunt identice (x şi y indică acelaşi punct)

2. Simetria

$d(x,y) = d(y,x)$.

Distanța între două puncte este aceeaşi indiferent de "direcție".

3. Inegalitatea triunghiului

$d(x,z) \leq d(x,y) + d(y,z)$

Distanta intre două puncte este distanța minimă în raport cu orice cale.

O asemenea funcție de distanță este cunoscută ca şi **metrică**. Împreună cu setul formează un **spațiu metric**.

II. Câteva distanțe uzuale

În cele ce urmează vom considera spațiul vectorial R^n şi vom nota x_i respectiv y_i doi vectori

$$x = x_1, x_2, ..., x_n$$

$$y = y_1, y_2, ..., y_n$$

1. Distanța Minkovski de ordin p

În spațiul vectorial R^n una din cele mai importante distanțe este **distanța Minkovski de ordin p** definită astfel

$$L_p = \left(\sum_{i=1}^{n} \left| x_i - y_i \right|^p \right)^{1/p}$$

p nu trebuie să fie obligatoriu întreg dar trebuie să nu fie mai mic ca 1 pentru a îndeplini inegalitatea triunghiului.

Dacă particularizăm obținem câteva distanțe foarte cunoscute şi importante:

2. **Distanța euclidiană** clasică

Corespunde situației p=2.

$$L_2 = \left(\sum_{i=1}^{n} \left| x_i - y_i \right|^2 \right)^{1/2} = \sqrt{\sum_{i=1}^{n} \left(x_i - y_i \right)^2}$$

Intuitiv reprezintă distanța între două puncte măsurată cu ruleta.

3. **Distanța Manhattan** (City Block)

Corespunde situației p=1.

$$L_1 = \sum_{i=1}^{n} \left| x_i - y_i \right|$$

Intuitiv reprezintă distanța parcursă în Manhattan între două puncte (unde avem de a face cu o rețea de străzi paralele pe direcțiile N-S şi respectiv E-V. În „limbajul locului”, reprezintă câte blocuri trebuie să mergem dintr-un punct în altul – de exemplu „walk five blocks”).

4. Distanța Cebîşev

Corespunde situației p = ∞

$$L_\infty = \lim_{p->\infty} \left(\sum_{i=1}^{n} \left| x_i - y_i \right|^p \right)^{1/p} = \max_i \left| x_i - y_i \right|$$

Intuitiv este distanța pe care trebuie să o străbată un rege pe tabla de şah pentru a se deplasa dintr-un câmp în altul.

Cazurile cu p diferit de 1, 2 şi ∞ sunt foarte rar folosite (nu prezintă interes).

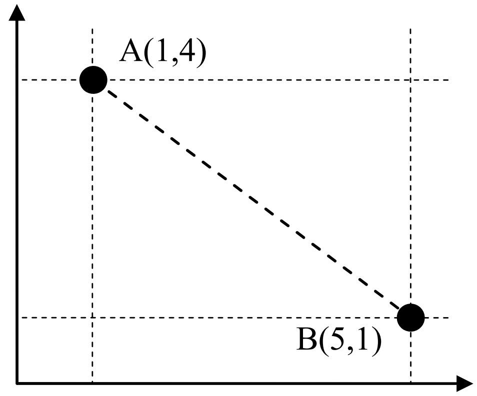

Figura 1. Exemplu evaluare distanțe

În Figura 1 se consideră cazul spațiului bidimensional şi două puncte A şi B (2 vectori). Evaluând formulele anterioare obținem pentru diferitele distanțe:

$L_2(A,B) = 5$

$L_1(A,B) = 7$!!!! :)

$L_\infty(A,B) = 4$

Alte distanțe de interes în ştiința calculatoarelor sunt **distanța Hamming** (definită ca numărul de componente ale vectorului care diferă) şi **distanța cosinus** (definită ca şi cosinusul unghiului dintre vectori).

III. Compresie near-lossless - introducere

În zilele noastre este o evidentă cerință pentru compresia de date şi, în mod special, pentru compresia de imagini. Deoarece metodele în care nu este acceptată nici o eroare de refacere (metode numite fără pierderi - "**lossless**") dau rezultate slabe pe imagini, pentru a obține rate de compresie mai mari trebuie să acceptăm existența unor erori în imaginea decomprimată. Aceste metode (numite cu pierderi – "**lossy**") au devenit populare odată cu dezvoltarea domeniului multimedia.

Criteriul de minimizat în compresia lossy a imaginii este, în abordarea clasică, cel dat de L_2, distanța euclidiană clasică dintre imaginea originală (necompresată) şi imaginea refăcută (decompresată). Suntem deci interesați de eroarea de la **nivelul întregii imagini**.

O altă clasă de aplicații este aceea în care este foarte important a avea o limită superioară a erorii la nivelul fiecărui pixel. Criteriul de minimizat este cel dat de distanța Cebîşev L_∞. În acest caz suntem interesați de eroarea la **nivel de pixel**.

Prima abordare (bazată pe L_2) este cea clasică şi în timpul anilor s-a investit mult efort de cercetare în această problemă. Cea mai populară metodă este cea bazată pe DCT şi standardizată în JPEG. Principalul dezavantaj al acestei metode este că nu putem avea nici o garanție asupra erorii la nivel de pixel. În unele aplicații (cum ar fi imaginile satelit, imaginile medicale, imagini cu scris etc.) este nevoie de garanții la nivel de pixel deoarece artefactele introduse prin compresie sunt inacceptabile (fără o limită superioară) în special pentru prelucrări automate ulterioare. De aceea în aceste domenii compresia de imagini cu pierderi a fost foarte rar acceptată (sau nu a fost acceptată deloc).

În ultimii ani interesul pentru metode bazate pe L_∞. a crescut rapid şi în consecință domeniul a solicitat investiție în cercetare. Deoarece de obicei suntem interesați doar în valori mici ale limitei erorii acceptate (±1, ±2, …, ±10) aceste metode poartă denumirea de metode **cu pierderi controlate** (“**near-lossless” NL)**. Evident, dacă limita erorii acceptate devine 0 regăsim situația clasică a compresiei fără pierderi (“lossless”).

Este foarte interesant de observat cum o schimbare mică, doar în măsura distanței folosite, ne pune în fața unui domeniu complet nou, rezultatele anterioare (bazate pe L_2) devenind complet irelevante (pentru L_∞). Este dovedit în literatura de specialitate că determinarea celei mai bune reprezentări (şi compresii) în limita erorii acceptabile ±k este o problemă NP-completă.

IV. Arhitectura predictivă pentru compresie near-lossless

Compresia near-lossless permite o compresie cu pierderi dar fără pierderea detaliilor. Deşi nu permite refacerea perfectă a imaginii, obține rezultate foarte bune ce se apropie ca performanțe de compresia lossy. În cazul compresiei predictive a imaginilor caracterul near-lossless se obține de obicei prin

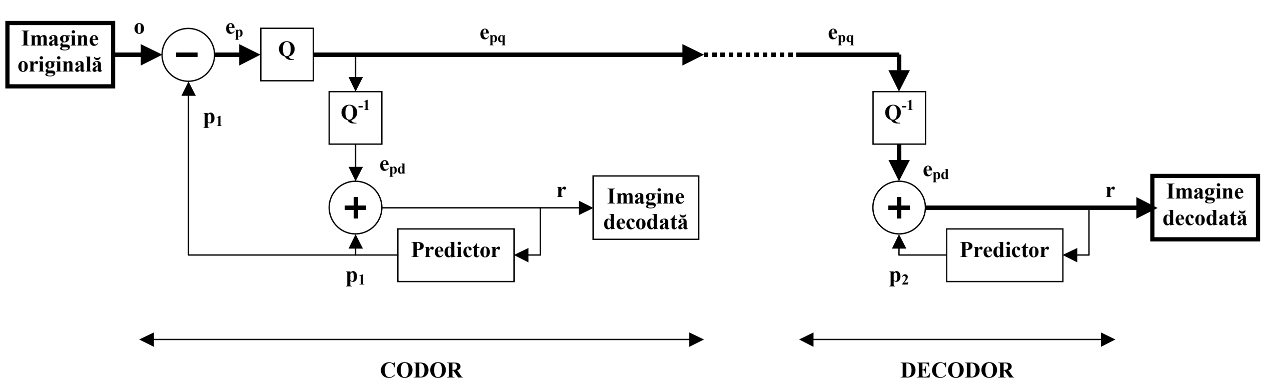

Figura 2. Arhitectura predictivă near-lossless

<u>cuantizarea</u> erorii de predicție.

Majoritatea metodelor de compresie lossless predictive au şi o variantă near-lossless, cu algoritmul de compresie fiind aproape identic ca implementare. Diferența între cele două metode este introducerea cuantizării în pasul ce precede transmiterea valorilor către codorul entropic. Toate considerentele de la codarea predictivă lossless a imaginilor (legate de pixelii pe baza cărora se poate face predicția, de efectul unei predicții mai bune sau mai rele asupra performanțelor, etc.) rămân în continuare valabile.

În Figura 2 este prezentată arhitectura predictivă near-lossless tipică. Cu notațiile din figura respectivă avem relațiile:

$$e_p = o - p_1 \quad \Rightarrow \quad o = e_p + p_1$$

$$r = e_{pd} + p_2$$

rezultă

$$r - o = (p_2 - p_1) + (e_{pd} - e_p)$$

Dacă facem ca $p_2 = p_1$ (predicția să fie identică la codor şi decodor) observăm că eroarea introdusă prin codare-decodare este egală cu eroarea introdusă prin procesul de cuantizare-decuantizare.

Valoare originală		Valoare cuantizată		Valoare decuantizată	Eroare introdusă
. . . . .		. . . . .		. . . . .	
7		1		5	-2
6		1		5	-1
5	=>	1	=>	5	0
4		1		5	1
3		1		5	2
2		0		0	-2
1		0		0	-1
0	=>	0	=>	0	0
-1		0		0	1
-2		0		0	2
-3		-1		-5	-2
-4		-1		-5	-1
-5	=>	-1	=>	-5	0
-6		-1		-5	1
-7		-1		-5	2
. . . . .		. . . . .		. . . . .	

Tabelul 1 Exemplificare a funcționării cuantizării

Cea mai populară variantă de cuantizare / decuanizare foloseşte următoarea regulă:

$$\dot{x} = Q^{-1}(Q(x)) = \left\lfloor \frac{x+k}{2\times k+1} \right\rfloor (2\times k+1) \qquad (1)$$

unde x este eroarea de predicție, k este maximul erorii de reconstrucție admise pentru fiecare pixel iar $\lfloor \ \rfloor$ reprezintă rotunjire la cel mai mare întreg mai mic sau egal. Codorul generează valoarea în acord cu:

$$Q(x) = \left\lfloor \frac{x+k}{2\times k+1} \right\rfloor \qquad (2)$$

valoarea transmisă decodorului iar eroarea de predicție este reconstruită conform:

$$Q^{-1}(Q(x)) = Q(x)\times(2\times k+1) \qquad (3)$$

Deoarece predicția trebuie să fie identică la codor şi decodor şi deoarece în acest caz, al compresiei near-lossless, avem diferențe între imaginea inițială (disponibilă la codor) şi imaginea decodată (disponibilă la decodor) codorul construieşte şi el o **copie a imagini decodate** (la fel ca decodorul). **Predicția** se face **pe baza imaginii decodate** atât la codor cât şi la decodor, obținând astfel predicții identice (Figura 2.). După cum s-a arătat eroarea care apare în imaginea decodată este exact eroarea care apare în procesul de cuantizare, nu apare o cumulare a erorilor predicționate. În Tabelul 1 prezentăm câteva situații de funcționare a cuantizorului (pentru k=2) din care se observă că eroarea introdusă prin cuantizare este în plaja +/- 2 (în limita criteriului near-lossless admis).

V. Alte abordări near-lossless simple

O modalitate de a obține o compresie near-lossless este de a cuantiza imaginea înaintea unei codării lossless. După realizarea cuantizării şi mapării aşa cum este ea definită în ecuația (2), imaginea rezultată are un domeniu dinamic sensibil mai redus şi deci va fi compresată mai tare comparativ cu situația inițială. Imaginea reconstruită ce întruneşte criteriul near-lossless poate fi obținută după decompresie şi remaparea valorii intensităților conform ecuației (3).

O altă cale de a asigura compresia near-lossless este în conjuncție cu o tehnica lossy. Aceasta poate fi realizată ca o tehnică în două etape, una lossy şi una near-lossless. O reprezentare lossy este realizată şi transmisă. După această, este trimisă varianta compresată near-lossless a diferențelor dintre original şi imaginea lossy. Cele mai performanțe se obțin de obicei pentru compresii inițiale lossy puternice.