

Digital Image Processing

23 November 2006

Dr. ir. Aleksandra Pizurica

Prof. Dr. Ir. Wilfried Philips

Aleksandra.Pizurica @telin.UGent.be

Tel: 09/264.3415



Image transforms

Introduction to image transforms

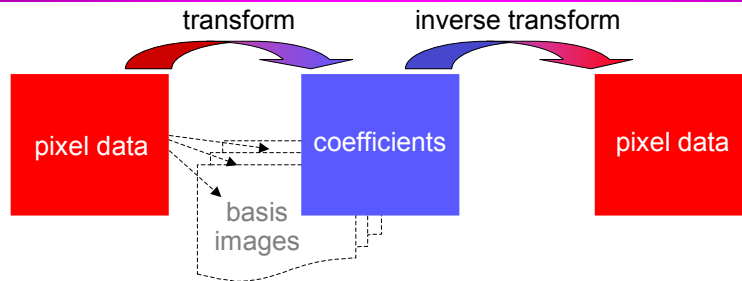


Image transform decomposes an image into **basis images**. Coefficients can be seen as weighting factors for those different “image components”

Important for compression, restoration and analysis. **For example:**

- **Compression:** define image transform such to “pack” the essential information about the image into **as few big coefficients as possible**
- **Restoration:** define image transform such that **biggest image coefficients indicate image edges**
- Other requirements can be defined for specific applications/problems

Series expansions: bases and frames

A series expansion of a signal f from space S is

$$f = \sum_i c_i \phi_i \quad (1)$$

where the elementary **atoms** or **building blocks** ϕ_i are **simple waveforms**

The set $\{\phi_i\}_{i \in \mathbb{Z}}$ is **complete** for S if **all** f from S can be decomposed as in (1)

A **complete and linearly independent** set $\{\phi_i\}_{i \in \mathbb{Z}}$ is **basis**

↳ yields a **non-redundant** representation of f

If the set $\{\phi_i\}_{i \in \mathbb{Z}}$ is **complete but not linearly independent** it is a **frame**

↳ yields a **redundant** representation of f

Non-redundant image transforms

Each image $b(x,y)$ of MN pixels can be written as a linear combination of exactly MN linearly independent basis images $p_i(x,y)$

$$b(x,y) = \sum_{i=0}^{MN-1} a_i p_i(x,y) \quad x=0 \dots M-1, y=0 \dots N-1$$

$\uparrow$ MN unknown MN equations

		x			
	y	0	1	2	$M-1$
0					
1					
2			$b(x,y)$		
$N-1$					

For given base functions $p_i(x,y)$ we compute a_i for a given image $b(x,y)$ by solving the set of equations

The set of equations has a unique solution for each image $b(x,y)$

$\Leftrightarrow$ the basis functions are linearly independent

$$\Leftrightarrow \sum_{i=0}^{MN-1} c_i p_i(x,y) = 0 \quad x=0 \dots M-1, y=0 \dots N-1 \quad \text{Holds only if all } c_i=0$$

Problem: $0.66(MN)^3$ computations are needed
too slow for practical images (b.v. $M=N=256$)

02b.5

Beeldtransformaties in vectornotatie

De coëfficiënten a_i bepalen éénduidig het beeld $b(x,y)$

De omzetting $b(x,y) \rightarrow a_i$ heet een lineaire transformatie

Soms is het nuttig de pixels van het beeld en de coëfficiënten te rangschikken in kolomvectoren $\mathbf{b}$ en $\mathbf{a} \Rightarrow$ **inverse** transformatie wordt $\mathbf{b} = P\mathbf{a}$

$$P^t = \begin{pmatrix} p_0(0,0) & \cdots & p_0(M-1,0) & p_0(0,1) & \cdots & p_0(M-1,1) & \cdots & p_0(M-1,N-1) \\ p_1(0,0) & \cdots & p_1(M-1,0) & p_1(0,1) & \cdots & p_1(M-1,1) & \cdots & p_1(M-1,N-1) \\ \vdots & & \vdots & \vdots & & \vdots & & \vdots \\ p_{MN-1}(0,0) & \cdots & p_{MN-1}(M-1,0) & p_{MN-1}(0,1) & \cdots & p_{MN-1}(M-1,1) & \cdots & p_{MN-1}(M-1,N-1) \end{pmatrix}$$

$$\mathbf{b} = \begin{pmatrix} b(0,0) \\ \vdots \\ b(M-1,0) \\ b(0,1) \\ \vdots \\ b(M-1,1) \\ \vdots \\ b(M-1,N-1) \end{pmatrix} \quad \mathbf{a} = \begin{pmatrix} a_0 \\ \vdots \\ a_{MN-1} \end{pmatrix}$$

02b.6

Orthogonale transformaties

Bij een orthogonale transformatie

- is de basismatrix P **orthogonaal**: $P^t P = I$
- en zijn de basisfuncties $p_i(x, y)$ **orthonormaal**:

$$\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} p_i(x, y) p_j(x, y) = \delta_{i,j} = \begin{cases} 0 & \text{als } i \neq j \\ 1 & \text{als } i = j \end{cases}$$

Voordeel: berekenen coëfficiënten vereist geen stelsel oplossen:

$$\mathbf{b} = P\mathbf{a} \Rightarrow P^t \mathbf{b} = P^t P \mathbf{a} = \mathbf{a} \text{ of dus: } \mathbf{a} = P^t \mathbf{b}$$

$$a_i = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} p_i(x, y) b(x, y)$$

$\Rightarrow$ veel sneller, maar nog steeds veel te traag voor praktische toepassingen op beelden van realistische afmetingen

$$\text{Parseval geldt (cfr. Fouriertransformatie): } E = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} b^2(x, y) = \sum_{i=0}^{MN-1} a_i^2$$

$\Rightarrow (a_i)^2$ is de bijdrage van basisfunctie $p_i(x, y)$ in de beeldenergie

02b.7

Projections onto basis images

$$b(x, y) = \sum_{i=0}^{MN-1} a_i p_i(x, y) \quad x = 0 \dots M-1, y = 0 \dots N-1$$

$$a_i = \langle B, P_i \rangle$$

$$\langle B, P \rangle = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} b(m, n) p^*(m, n) \quad \text{inner product}$$

The transform coefficient a_i is simply the inner product of the i -th basis image with the given image.

It is also called projection of the image on the i -th basis image

02b.8

Eigenschappen orthogonale transformaties

Voordelen van orthogonale transformaties

- snellere berekening coëfficiënten a_i voor een gegeven beeld $b(x,y)$
- eenvoudige kleinste-kwadratenbenadering = de beste benadering van $b(x,y)$ met $n < MN$ basisfuncties $p_i(x,y)$
 $\Rightarrow$ deze bekom je door de coëfficiënten a_i van de weggelaten basisfuncties 0 te stellen en de andere a_i te behouden

$$a_i = \underset{c_i}{\operatorname{argmin}} \left\| b(x,y) - \sum_{i=0}^{n-1} c_i p_i(x,y) \right\| \quad \text{met} \quad \|f(x,y)\|^2 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} |f(x,y)|^2$$

Bij niet-orthogonale basisfuncties hangen de optimale coëfficiënten af van n , het aantal basisfuncties in de benadering

- gevolg van de stelling van Parseval: de totale kwadratische afwijking tussen twee beelden $b(x,y)$ en $b'(x,y)$ kan worden beoordeeld adhv. hun coëfficiënten:

$$\|b(x,y) - b'(x,y)\|^2 = \sum_{i=0}^{MN-1} |a_i - a'_i|^2$$

02b.9

Eigenschappen orthogonale transformaties

Ongecorrleerde ruis (b.v. witte gaussiaanse ruis) op een beeld wordt omgezet in ongecorrleerde afwijkingen op de coëfficiënten

- addieive ruis $b'(x,y) = b(x,y) + n(x,y) \Rightarrow a'_i = a_i + \varepsilon_i$

$$E(n(x,y)n(x',y')) = \begin{cases} 0 & \text{als } x' \neq x \text{ of } y' \neq y \\ \sigma^2 & \text{als } x'=x \text{ en } y'=y \end{cases}$$

$$\Rightarrow E(\varepsilon_i \varepsilon_j) = \begin{cases} 0 & \text{als } i \neq j \\ \sigma^2 & \text{als } i = j \end{cases}$$

02b.10

Scheidbare orthogonale transformaties...

We kunnen de basisfuncties ook met een dubbele index noteren:

$p_i(x,y)$ met $0 \leq i < MN \rightarrow p_{i,j}(x,y)$ met $0 \leq i < N$ en $0 \leq j < M$

Bij een **scheidbare** orthogonale transformatie zijn de

basisfuncties van de vorm: $p_{i,j}(x,y) = q_j(x)r_i(y)$

$$\Rightarrow b(x,y) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{i,j} p_{i,j}(x,y) = \sum_{i=0}^{N-1} r_i(y) \sum_{j=0}^{M-1} a_{i,j} q_j(x)$$

$$a_{i,j} = \sum_{y=0}^{N-1} r_i(y) \underbrace{\sum_{x=0}^{M-1} b(x,y) q_j(x)}_{\substack{2M \text{ berekeningen voor gegeven } x \text{ en } j \\ \Rightarrow 2M^2N \text{ berekeningen in totaal}}}$$

$\underbrace{\hspace{10em}}_{\substack{2N \text{ berekeningen voor gegeven } i \text{ en } j \text{ en als de binnenste} \\ \text{som al is berekend} \Rightarrow 2MN^2 \text{ berekeningen in totaal}}}$

Stel $M=N \Rightarrow$ Een scheidbare transformatie vraagt $4N^3$ berekeningen per beeld, d.w.z. $4N$ bewerkingen per pixel; voor een niet-scheidbare transformatie is dit $2N^4$ per pixel en $2N^2$ per pixel

Ook de inverse transformatie is sneller in het scheidbaar geval (**controleer**)

02b.11

...Scheidbare orthogonale transformaties

Een scheidbare orthogonale transformatie $\mathbf{a} = P^t \mathbf{b}$ is equivalent met

- eerst een **orthogonale** transformatie Q^t toepassen op elke rij van het beeld
- dan een **orthogonale** transformatie R toepassen op elke kolom van het resultaat
- of in omgekeerde volgorde

$$\underbrace{a_{i,j}}_{\substack{\uparrow \\ N \times M \text{ matrix } A}} = \sum_{y=0}^{N-1} \underbrace{r_i(y)}_{\substack{\uparrow \\ N \times N \text{ matrix } R^t}} \sum_{x=0}^{M-1} \underbrace{b(x,y)}_{\substack{\uparrow \\ N \times M \text{ matrix } B}} \underbrace{q_j(x)}_{\substack{\uparrow \\ M \times M \text{ matrix } Q}}$$

$$B_{y,x} = b(x,y) \text{ en } Q_{x,j} = q_j(x) \Rightarrow \sum_{x=0}^{M-1} b(x,y) q_j(x) = \sum_{x=0}^{M-1} B_{y,x} Q_{x,j} = (BQ)_{y,j}$$

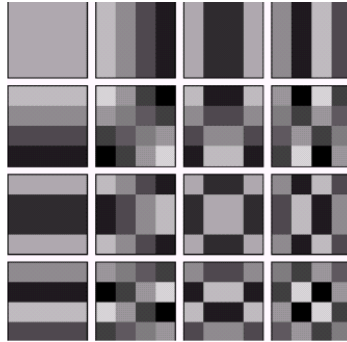
$$R_{y,i} = r_i(y)$$

$$\mathbf{a} = P^t \mathbf{b} \Leftrightarrow \mathbf{a} = R^t B Q \quad \text{en} \quad \mathbf{b} = P \mathbf{a} \Leftrightarrow \mathbf{b} = R A Q^t$$

02b.12

Discrete Cosine Transform

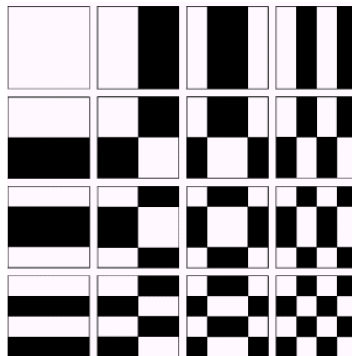
- The cosine transform has excellent energy compaction for highly correlated data
- Often used in compression (JPEG)



02b.13

Walsh-Hadamard Transform

- Walsh-Hadamard basis functions are **binary**
- Faster than (co-)sinusoidal transforms – no multiplications involved
- Useful in digital **hardware** implementations
- Applications in **image compression** and **design of codes** (Hadamard coefficients of a sequence – bit reversed version of the Gray code of that sequence)



02b.14

Excercise

