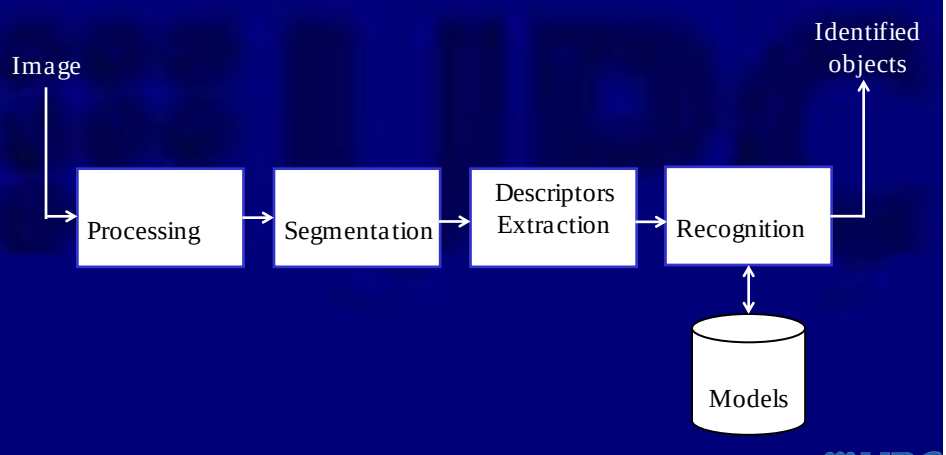
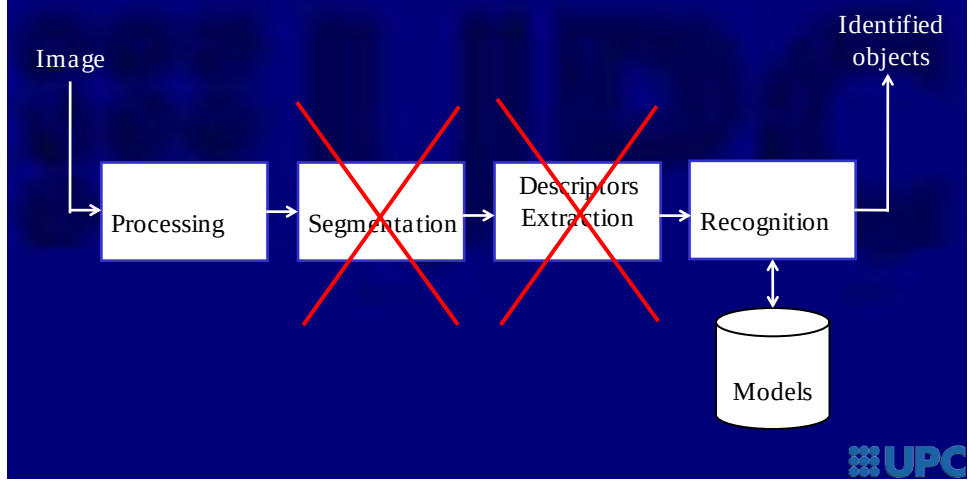
Local Features

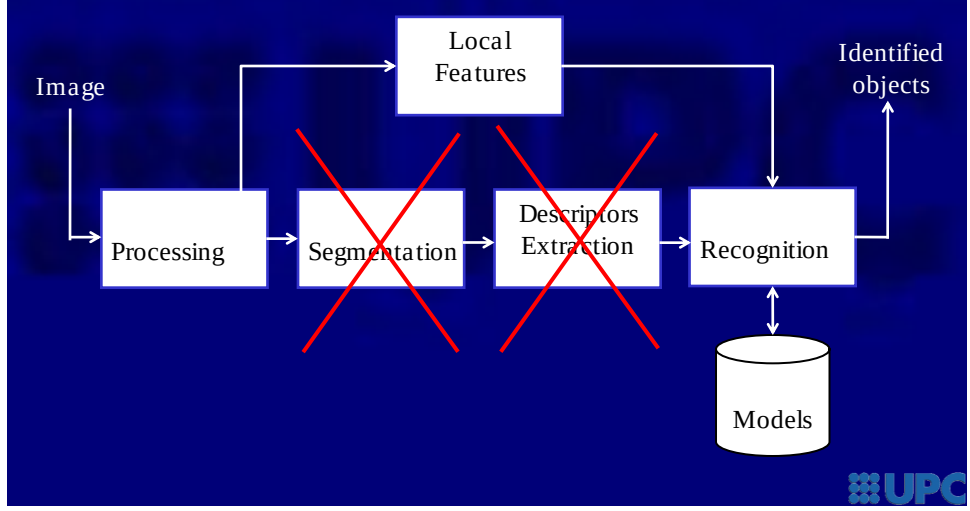
A computer vision system



A computer vision system



A computer vision system



Local Features

- Les característiques globals són sensibles al soroll i les oclusions.
- L'alternativa és usar característiques locals:
  - Si algunes característiques queden ocluïdes, l'objecte potser es podrà identificar a partir d'altres característiques que siguin visibles.

UPC

Corners are Simple Features

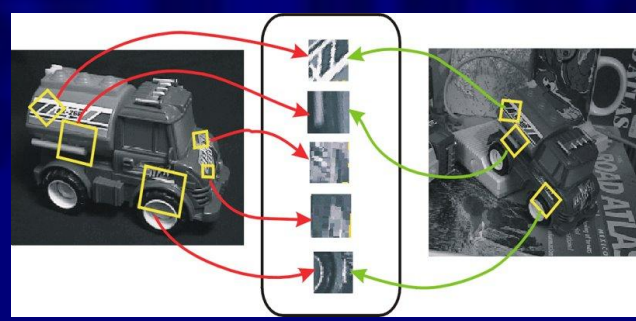
↘ Can you see common feature point between the two images?

UPC

Invariant local features

Buscar característiques que siguin invariants a transformacions diverses:

- geometric invariance: traslació, rotació, escala...
- photometric invariance: il.luminació, color...



Feature Descriptors

Features "bones"

- Discriminatives (Saliency)
Han de ser significativament diferents i distingibles d'altres features properes.

- Repetitivitat
S'han de poder localitzar en imatges diferents, encara que aquestes presentin transformacions.

- Compactes
Poques dades, molta informació. Eficiència computacional

Aplicacions

- Image alignment and stitching
- Image retrieval
- Motion tracking
- Robot navigation
- Face detection
- Object recognition
- Gesture recognition
- Human action understanding
- Biometric identification

Aplicacions. Exemples

Image Stitching (to generate panorama)

Aplicacions. Exemples

Local Features

- Histogrames
- Transformada de Hough
- Vèrtexos
- Scale Invariant Feature transform (SIFT)
- Haar Features (face detection)

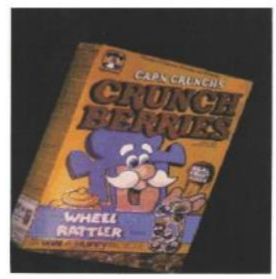
Histogrames

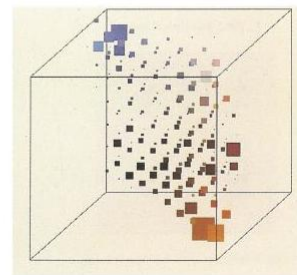
Histogrames de color

- Els histogrames són un tipus de descriptor basats en aparença
- El color és invariant a transformacions geomètriques
- El color és una característica local. Està definit per cada píxel
- Usarem l'histograma de color com una descripció estadística d'una regió o objecte
- Al ser un descriptor local, pot ser robust a oclusions

Histogrames de color

- Colour histograms are colour statistics
  - Here, RGB as an example
  - Given: tristimulus R, G, B for each pixel
  - Compute a 3D histogram
  - $h(R,G,B) = \#(\text{pixels with colour } (R,G,B))$





UPC

Histogrames de color

Colour Normalization

- One component of the 3D colour space is intensity
  - If a colour vector is multiplied by a scalar, the intensity changes but not the colour itself.
  - This means colours can be normalized by the intensity.
  - Note: intensity is given by $I = (R + G + B)/3$
  - Chromatic representation:

$$r = \frac{R}{R+G+B} \qquad g = \frac{G}{R+G+B} \qquad b = \frac{B}{R+G+B}$$

Since $r + g + b = 1$, only 2 parameters are needed to represent colour (knowing r and g, we can deduce $b = 1 - r - g$).

⇒ Can compute colour histogram using r, g, and b instead.

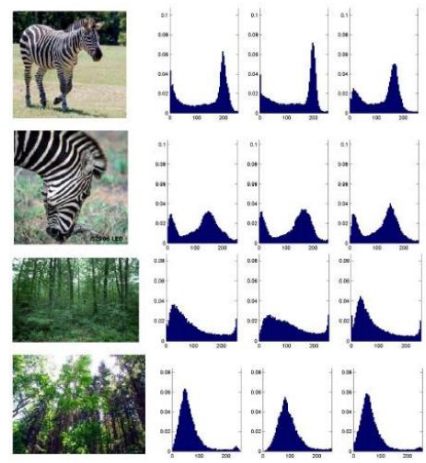
UPC

Descriptors basats en histogrames

Object Recognition based on Colour Histograms

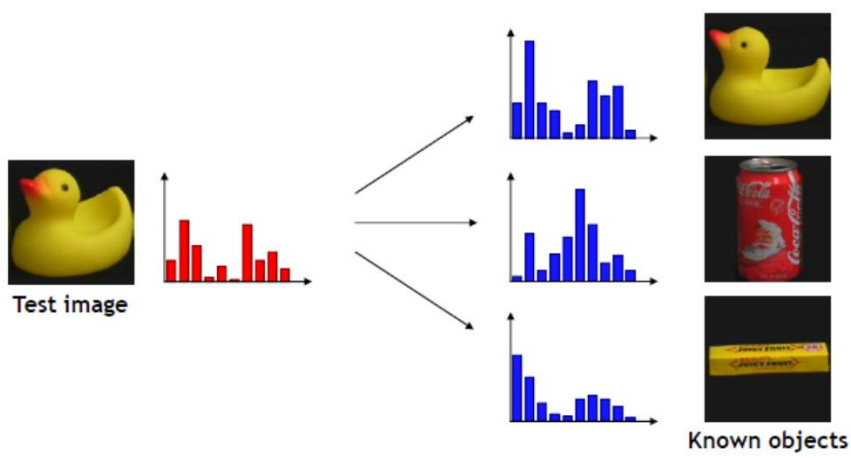
Colour histograms

- are discrete approximation of the colour distribution of an image.
- contain no spatial information ⇒ invariant to translation, scale, and rotation



UPC

Descriptors basats en histogrames



UPC

Object recognition using histograms

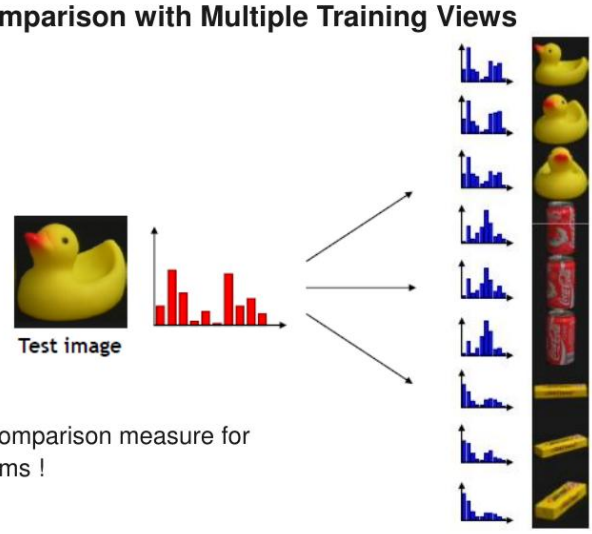
- Simple algorithm

1. Build a set of histograms $H = \{\boldsymbol{h}_i\}$ for each known object.
   - More exactly, for each view of each object.
2. Build a histogram $\boldsymbol{h}_t$ for the test image.
3. Compare $\boldsymbol{h}_t$ with each $\boldsymbol{h}_i \in H$ using a suitable histogram comparison measure.
4. Select the object with the best matching score;
   or reject the test image if no object is similar enough.

This is known as the "**nearest-neighbour**" strategy.

Object recognition using histograms

Histogram Comparison with Multiple Training Views

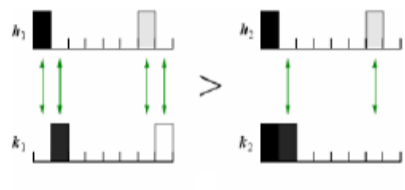


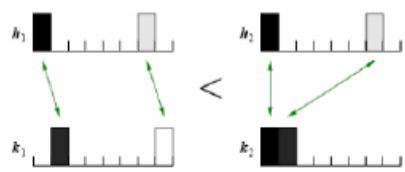
⇒ Need a good comparison measure for colour histograms !

Comparació d'histogrames

What is a Good Comparison Measure?

↘ How to define matching cost?





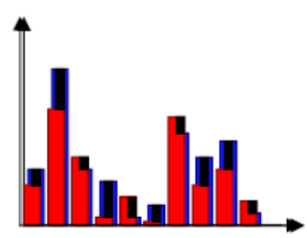
Comparació d'histogrames

Euclidean distance (L_2 norm)

$$d(\boldsymbol{q},\boldsymbol{v}) = \sum_i (q_i - v_i)^2$$

↘ Motivation of the Euclidean distance:

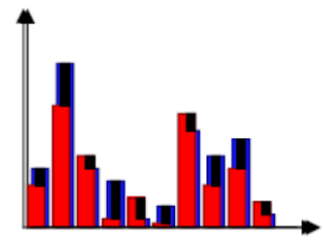
- Focuses on the differences between the histograms.
- Interpretation: distance in the feature space.
- Range: $[0,\infty)$.
- All cells are weighted equally.
- Not very robust to outliers !



Comparació d'histogrames

Chi-Square distance:

$$d(\boldsymbol{q}, \boldsymbol{v}) = \sum_i \frac{(q_i - v_i)^2}{q_i + v_i}$$



- Motivation of the χ^2 distance:
  - Statistical background
  - Test if two distributions are different.
  - Possible to compute a significance score.
  - Range: $[0, \infty)$.
  - Cells are not weighted equally !
  - More robust to outliers than the Euclidean distance, if the histograms contain enough observations…

Comparació d'histogrames

- Which measure is the best?
  - It depends on the application
  - Euclidean distance is often not robust enough.
  - Generally, χ^2 distance gives good performance for histograms
  - KL/Jeffreys divergence works well sometimes, but is expensive
  - EMD is the most powerful, but also very expensive.

Exemple: image retrieval

- The image retrieval problem concerns the retrieval of those images in a database that best match a query image.

L2 distance

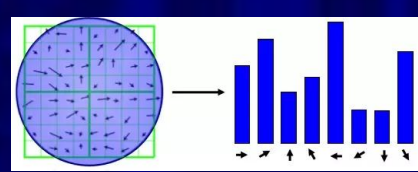
Jeffrey divergence

χ^2 distance

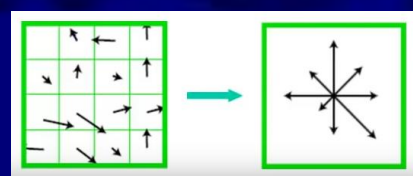
Earth Movers Distance

Histogrames d'orientacions

Idea: Podem descriure localment els objectes usant les direccions dels gradients. No cal localitzar els contorns, només saber la distribució de les seves orientacions

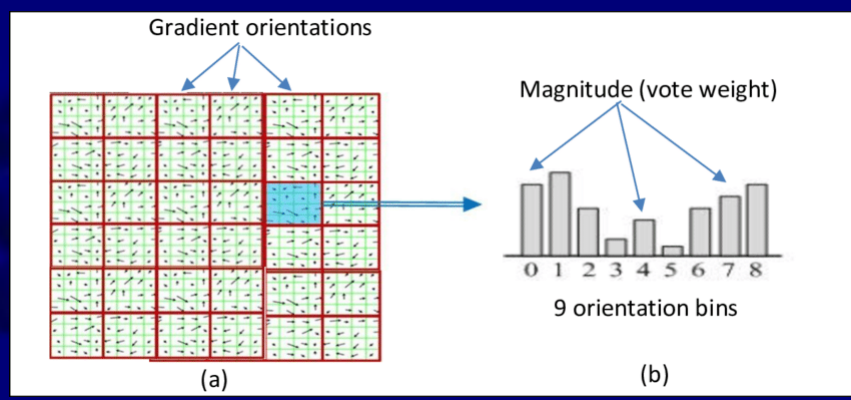


Representació linial



Representació cíclica

Histogrames d'orientacions



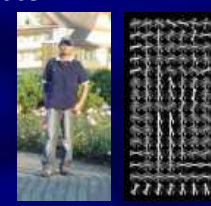
Múltiples subimatges - Múltiples histogrames

HOGs (Histograms of Oriented Gradients)

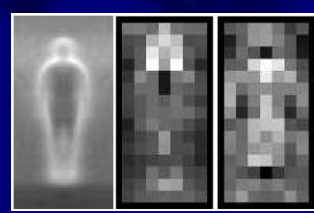
- Dalal-Triggs, 2005
- Divide image into small sub-images: “cells”
- Accumulate a histogram of edge orientations within that cell
- The combined histogram entries are used as the feature vector describing the object
- To provide better illumination invariance (lighting, shadows, etc.) normalize the cells across larger regions incorporating multiple cells: “blocks”

HOGs . Detecció de persones

1. Descriure cada mostra usant un HOG template



2. Entrenar una SVM com a classificador



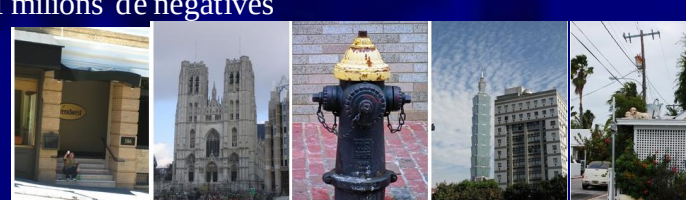
Code available: http://pascal.inrialpes.fr/soft/olt/

UPC

HOGs . Detecció de persones

Milers de mostres positives

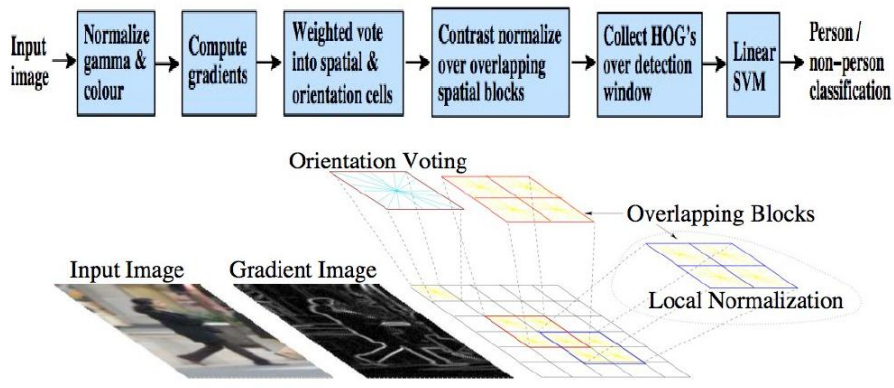
I milions de negatives



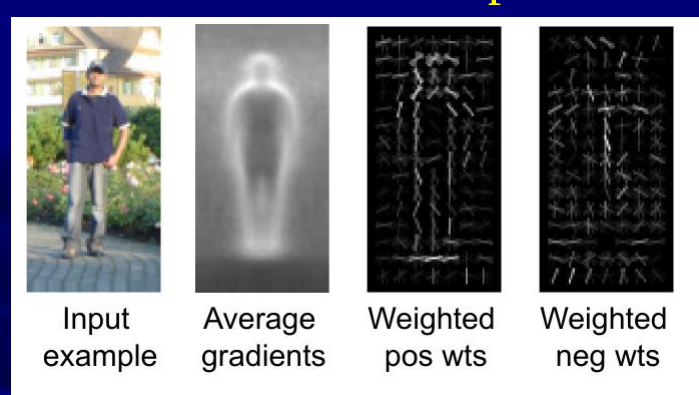
UPC

HOGs . Detecció de persones

A worked example: Dalal and Triggs CVPR'05



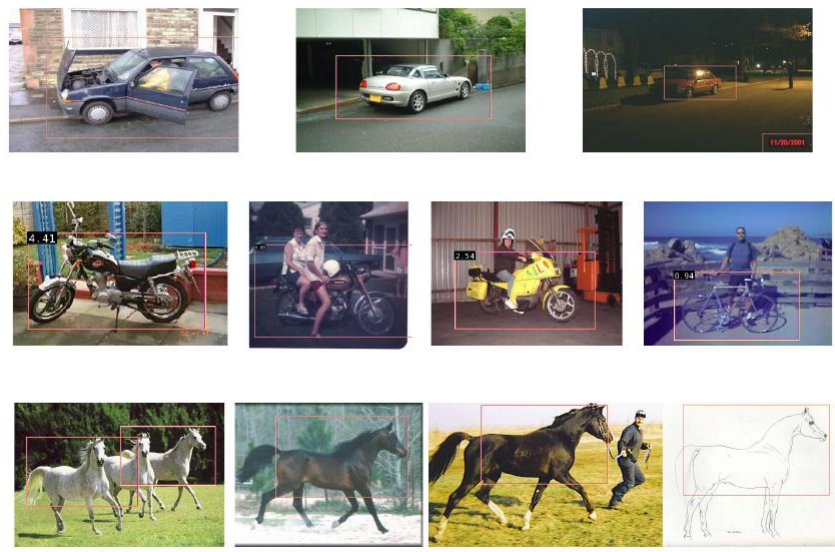
HOGs . Detecció de persones



Most important cues are head, shoulder, leg silhouettes

Vertical gradients inside a person are counted as negative

Applications to Other Classes



M. Everingham et al. *The 2005 PASCAL Visual Object Classes Challenge.* Proceedings of the PASCAL Challenge Workshop, 2006. 25

Transformada de Hough

Transformada de Hough

- Dissenyada originalment per a la detecció de rectes
- Molt robusta a soroll, imperfeccions i oclusions
- Cada recta en la imatge original es transforma en una punt en l'espai de Hough
-Aquest punt ve determinat per les coordenades (m,b) de la recta

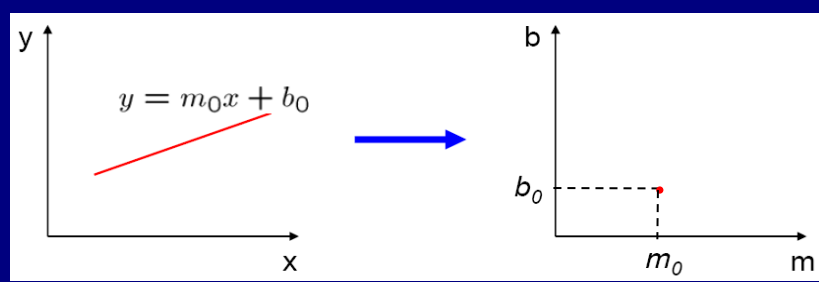


Image space

Hough parameter space

Mapejat en l'espai de Hough

- Donat un píxel (x_0,y_0), quantes rectes hi passen?

Resposta: totes les que satisfan l'equació $y_0 = mx_0 + b$
(això és una recta en l'espai de Hough)

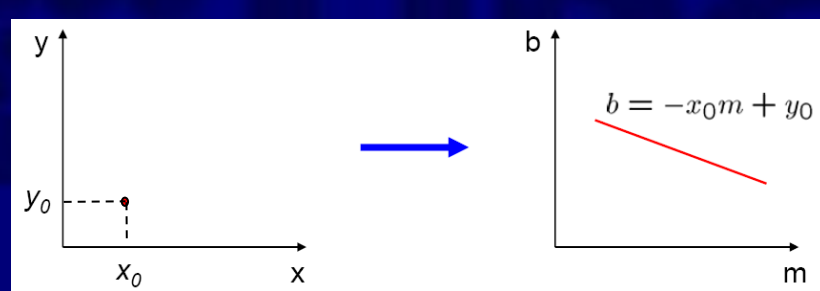


Image space

Hough parameter space

Mapejat en l'espai de Hough

- On es mapeja la recta que passa pels píxels (x_0,y_0) i (x_1,y_1) en l'espai de Hough?

Resposta: En la intersecció de les rectes

$b = -x_0 m + y_0$

$b = -x_1 m + y_1$

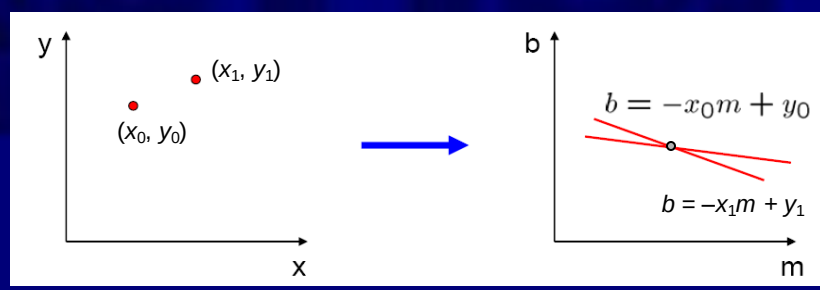


Image space Hough parameter space

Mapejat en l'espai de Hough

- Podem implementar l'espai de Hough com una taula 2D d'acumuladors
- La presència de molts píxels alineats en la imatge original (una recta) es traduirà com un pic en la taula de Hough
- La transformada de Hough no és res més que un sistema de votació

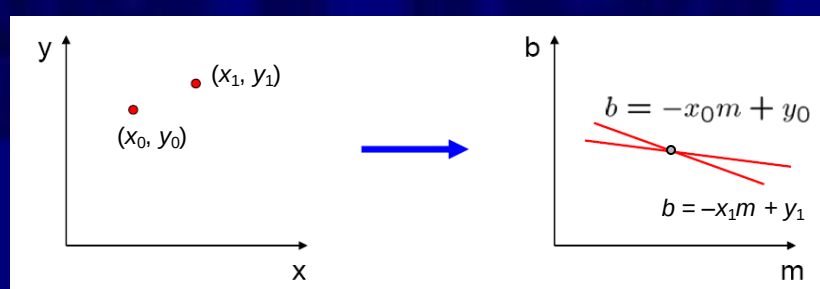


Image space Hough parameter space

Transformada de Hough. Algorisme

1. Crear una matriu de Hough (m,b) i dimensionar-la.
2. Cada posició de la matriu és un acumulador. Posar-los tots a 0.
3. Per a cada píxel (x,y) de la imatge d'entrada (imatge de contorns), incrementar les posicions de la matriu (m,b) que satisfan l'equació.
4. Els màxims de la matriu (m,b) es corresponen a la presència de rectes en la imatge.

- Si hi ha diverses rectes a la imatge, tindrem diversos pics a la matriu.

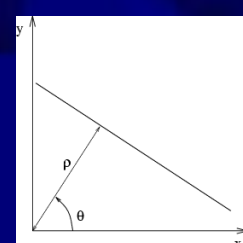
Dimensions de la taula de Hough

-Problema: afitar m i b.

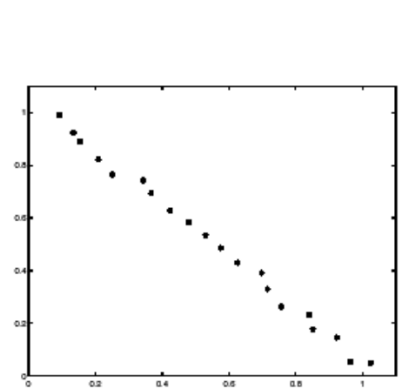
-Solució: S'usa la forma polar de la recta:

$$\rho = x \cos(\theta) + y \sin(\theta)$$

- ara els punts (x,y) queden mapejats com funcions sinusoïdals en l'espai (ρ,θ).
- ρ representa la llargada del vector normal desde l'origen de la imatge a la recta i θ la seva direcció



Taula de Hough (ρ,θ)



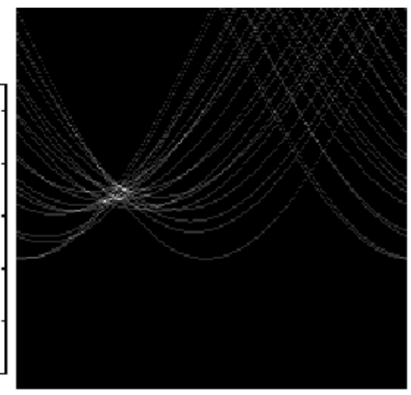
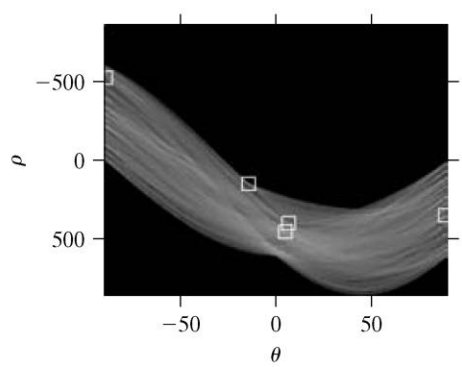
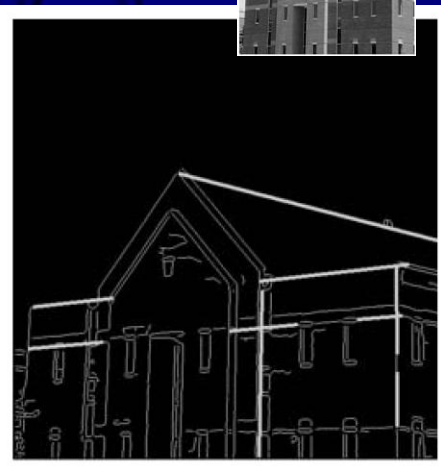


Image space

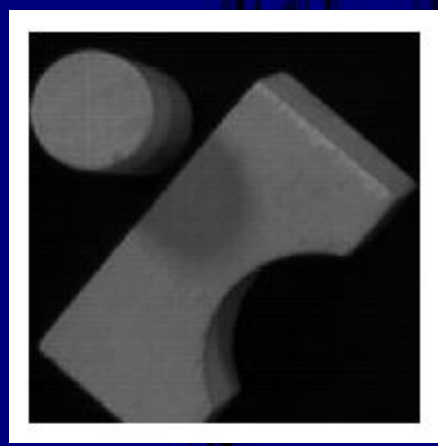
Votes
(Horizontal axis is θ, vertical is rho)

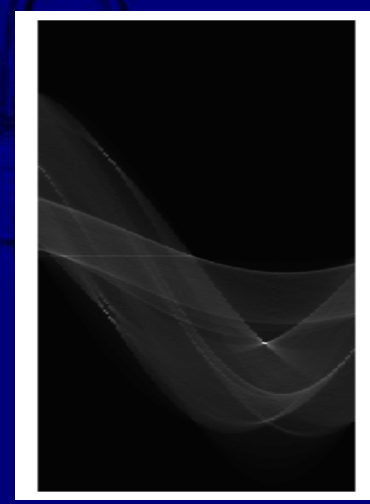
Exemple



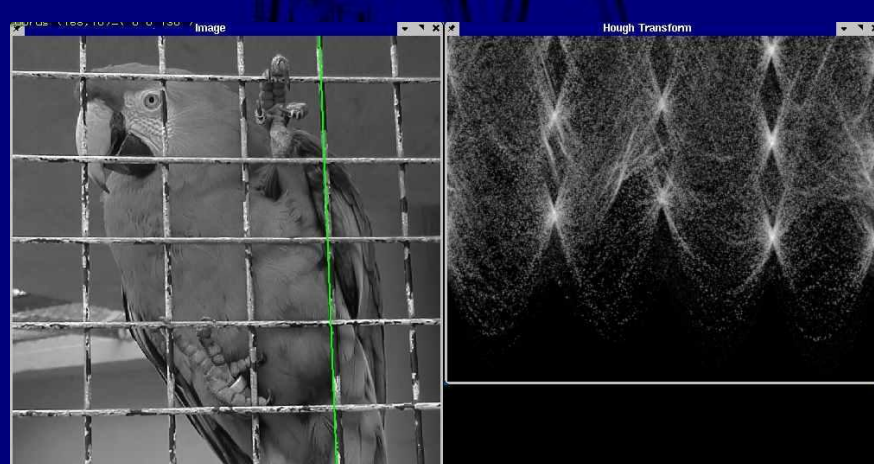


Exemple





Exemple



Efectes del soroll

- Els pics a la taula de Hough no són fàcils de localitzar

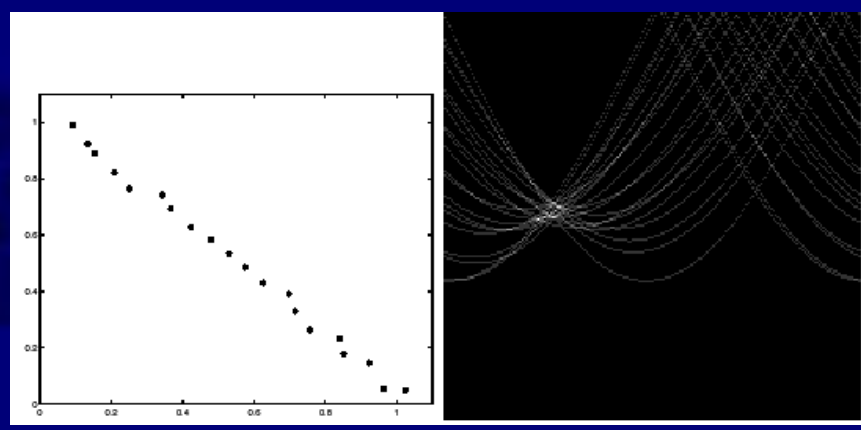


Image space

Hough parameter space

Transformada de Hough. Implementació

- Resolució de la taula. Quantes caselles? De quina mida?

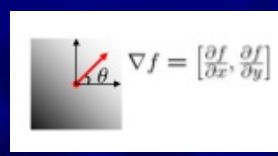
Si les fem massa grans, rectes diferents incrementaran el mateix acumulador
Si les fem massa petites, es fraccionaran rectes

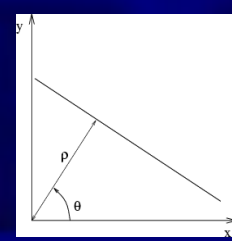
- Mai tindrem l'acumulació en una única casella. Això és per culpa del soroll en la imatge i la discretització de la taula de Hough.
Cal Considerar els pics molts propers com a una única recta.
Solucions: non-maxima supression, filtratge de la taula...

- Treballeu només amb píxels rellevants. Aquells que tenen un gradient important

Hough transform speed-up

- Coneixem la direcció del contorn (podem disposar d'aquesta informació. És la direcció del gradient)





- Podem fixar θ en la taula de Hough i incrementar un únic acumulador.

Speeded-up algorithm

1. Per a la imatge d'entrada, obtenir mòdul (G) i direcció (α) del gradient.

2. Crear la taula de Hough. Serà una matriu 2D d'acumuladors H(ρ,θ). Cal dimensionar-la correctament

3. Inicialitzar tots es acumuladors a 0

4. per a cada pixel G(x,y) > Thresh /* gradient prou gran */
   {
   θ = α(x,y) /* direcció del gradient */
   ρ = x cosθ + y sinθ
   incrementar H(ρ,θ)
   }

5. Buscar màxims locals a H(ρ,θ)

Transformada de Hough per a corbes

- La transformada de Hough es pot usar per a trobar qualsevol tipus de corba que es pugui expresar en forma paramètrica

$$y = f(x_1, x_2, \ldots, x_n)$$

- $x_1, x_2, \ldots, x_n$ serien els paràmetres de l'espai de Hough

- Si n és gran, la dimensio de la matriu de Hough és també gran i la complexitat es dispara.

Curse of dimensionality !!

Transformada de Hough per a trobar cercles

Finding Circles by Hough Transform

- Equation of Circle:

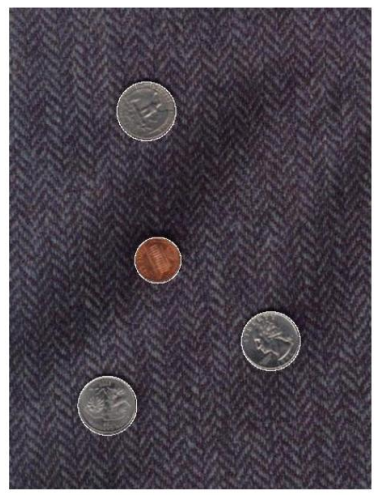
$$(x_i - a)^2 + (y_i - b)^2 = r^2$$

- If radius is known: (2D Hough Space)
  - Accumulator Array $A(a,b)$

- If radius is not known: 3D Hough Space!
  - Use Accumulator array $A(a,b,r)$

Exemple

Finding Coins



Note that because the quarters and penny are different sizes, a different Hough transform (with separate accumulators) was used for each circle size.