

Binary Symbol Recognition from Local Dissimilarity Map

GREC - 2009

F. Morain-Nicolier, J. Landré, S. Ruan

frederic.nicolier@univ-reims.fr

<http://pixel-shaker.fr>

CRESTIC - URCA/IUT Troyes

22 juillet 2009

- Symbol Recognition using raw pixel information
- Not a statistical approach : no learning
- Not a structural approach : no primitive

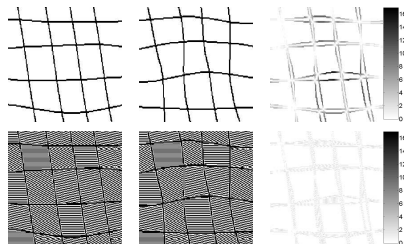
⇒ **Build a good (di)similarity measure between images**

Local to Global Dissimilarity

- Local Dissimilarity Map

$$\text{LDM}_{A,B} = |A - B| \max(\text{dt}_A, \text{dt}(p)) \quad (1)$$

$$= B \text{dt}_A + A \text{dt}_B. \quad (2)$$

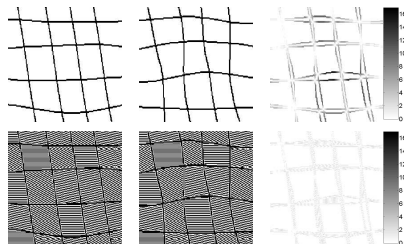


Local to Global Dissimilarity

- Local Dissimilarity Map

$$\text{LDM}_{A,B} = |A - B| \max(\text{dt}_A, \text{dt}(p)) \quad (1)$$

$$= B \text{dt}_A + A \text{dt}_B. \quad (2)$$



- Global Dissimilarity Measure : sum the (squared) values :

$$\text{GDM}(A, B) = \alpha \sum_p B(p) \text{dt}_A^2(p) + \beta \sum_p A(p) \text{dt}_B^2(p). \quad (3)$$

Local to Global Dissimilarity

- Why α and β ?
- Chamfer Matching [Borgefors] : sum of the distances of each translated model's pixel to the nearest images's pixel.
- **GDM is a double Chamfer Matching** :

$$\text{GDM}_{A,B} \sim \alpha \text{CS}(A, B) + \beta \text{CS}(B, A) \quad (4)$$

- CS : how is M similar to I ?
- GDM : + how is I similar to M ?
- **symetric or asymmetric matching** (links to psychological models of similarity [Tversky])?

First Algorithm

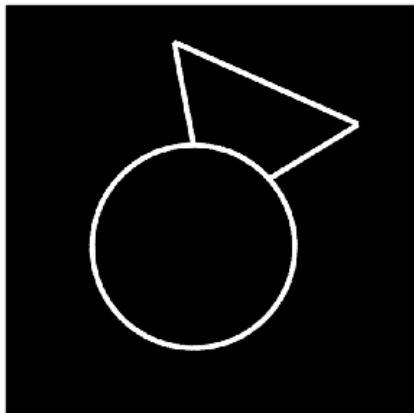
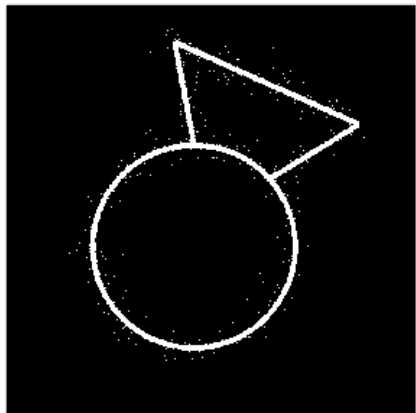
- 1 I contains an unknown symbol.
- 2 **Foreach model M :**
 - **compute the dissimilarity** $GDM(I, M)$ between I and M .
- 3 Keep the model with the **lowest dissimilarity as winner**.

First Algorithm

- 1 I contains an unknown symbol.
- 2 **Foreach model M :**
 - **compute the dissimilarity** $GDM(I, M)$ between I and M .
- 3 Keep the model with the **lowest dissimilarity as winner.**
 - Tested on the GREC2005 international symbol recognition contest database (electronic and architecture) : six degradations
 - 25 symbols to be recognized in 50 images.
 - $\alpha = 1$ and $\beta = 0$.

Results (no deformation)

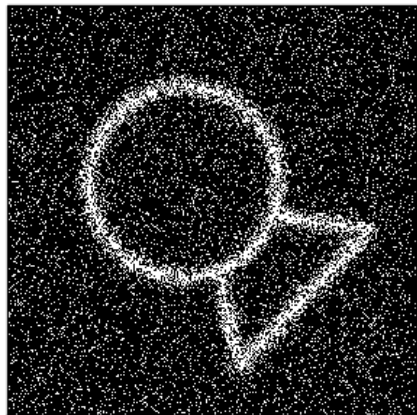
Degradation 1



100 %

Results (no deformation)

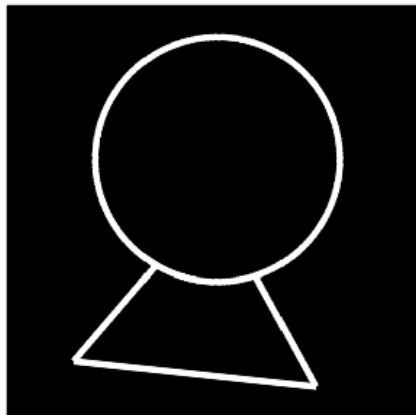
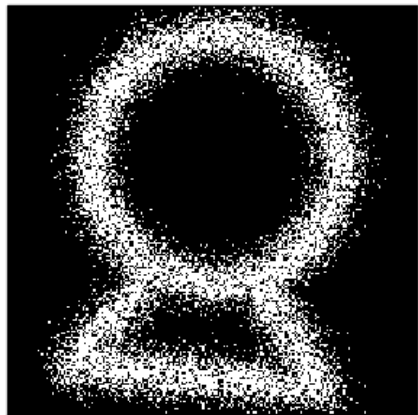
Degradation 2



100 %

Results (no deformation)

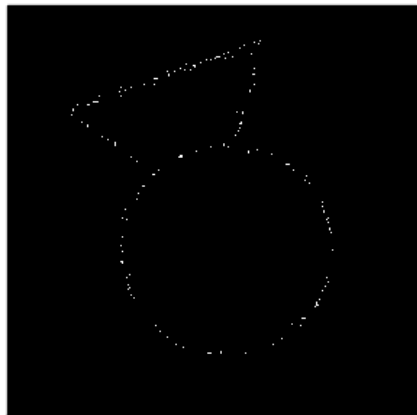
Degradation 3



100 %

Results (no deformation)

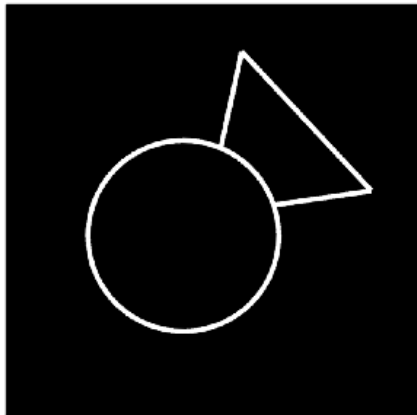
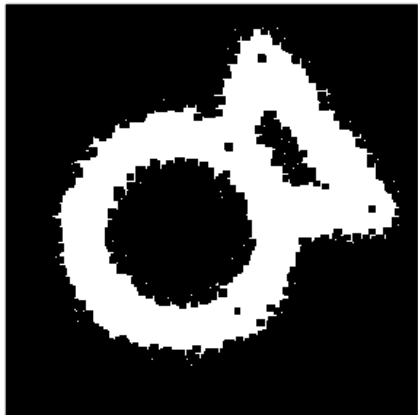
Degradation 4



100 %

Results (no deformation)

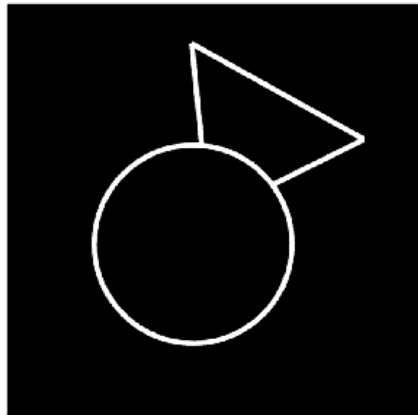
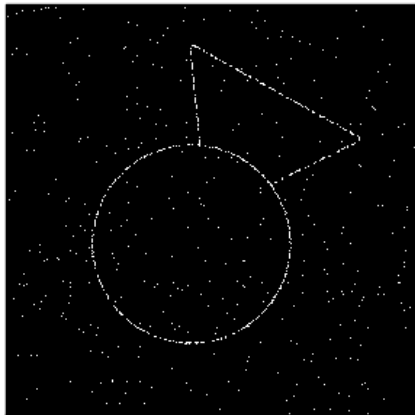
Degradation 5



100 %

Results (no deformation)

Degradation 6



54 % (but best is 59.81 %)

And with deformations?

- Transform **rotation and scale into translations**
⇒ log-polar transform

And with deformations ?

- Transform **rotation and scale into translations**
⇒ log-polar transform
- Given two images I and M to be registered :
 - **Compute I_{lp} and M_{lp}**
 - **Which translation produces the most similar images ?**
⇒ compute the global dissimilarity between I_{lp} and each translation of M_{lp} .
 - Very fast with cross-correlations (in Fourier Domain) :

$$LOC_{I,M} = \alpha dt_I^2 \star M + \beta I \star dt_M^2. \quad (5)$$

- **Find the minimum** of $LOC_{I,M}$ and find the rotation and scale parameters from its position.

Complete Algorithm

- 1 Compute I_{lp} , **log-polar representation of I** .
- 2 Foreach model M :
 - 1 compute M_{lp} , **log-polar representation of M**
 - 2 find the **global minimum** of $\text{LOC}_{I_{lp}, M_{lp}}$ (fast computation)
 - 3 find (σ, θ) the scaling and rotation parameters of M .
 - 4 compute $M_{\text{corrected}}$ from (ρ, θ) , by **applying reverse scale and rotation**.
 - 5 compute the dissimilarity $\text{GDM}(I, M_{\text{corrected}})$ between $M_{\text{corrected}}$ and I .
- 3 Keep the model with the **lowest dissimilarity as winner**.

Advantages and Drawbacks

rot/scale	m1	m2	m3	m4	m5	m6
without	100%	100%	100%	100%	100%	54%
with	96%	40%	94%	70%	42%	12%

- **Quite good performances** (lowered by 5%-10% with 100 images/150 symbols)
- **without any a-priori knowledge, pre-processing, nor primitive extraction**
- **Fast but still needs parameters estimations**
- Next :
 - Gray level images \Rightarrow Localization with rotation/scale variations.
 - Balance the asymetry, e.g. $\alpha = 0.9$ and $\beta = 0.1$ (take into account noise)
 - Direct measure in log-polar domain.
 - Apply preprocessing to boost performances.