Fast template matching and selection in the binary domain

Mathieu Delalandre (PhD)

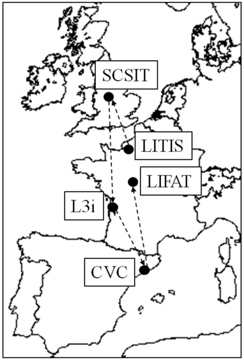
LIFAT Laboratory, RFAI group, Tours city, France
mathieu.delalandre@univ-tours.fr

July 12, 2016

Mathieu Delalandre - CV in short

PhD in Computer Science with 10 years of experience

2001-2005:	PhD, Rouen University, France
2005-2009:	Research Fellow Positions (UK, France, Spain)
2009-today:	Assistant Professor (LIFAT Lab, Tours city, France)



Mathieu Delalandre - CV in short

Ongoing activities on image processing (starting 2009)

Research topics	*Application domains*
Processing in the transform domain	Document image networking
Object detection and template matching	Comics / Manga copyright protection
Local descriptors and detectors	Symbol and logo detection and recognition

Past activities: image understanding, graph-based representation, performance evaluation

Mathieu Delalandre - CV in short

Publications: Journals (TIP, PR, PRL, IJDAR), Conferences Workshops (ICDAR, DAS, GREC).

Projects and funding investigations

2001-2009:	participation to 9 international and national projects.
2009-2015:	VIED P322 PhD scholarship, DOD project (483 k€), SATT CopyBD project (98 k€), JSPS research fellow
Ongoing:	ScannerLoire project (199 k€), BR PhD scholarship, VIED P165/P911 PhD scholarship

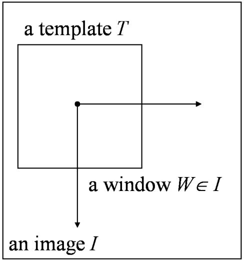
Mathieu Delalandre - CV in short

Partnerships:	HDU (Thanh Hoa, Vietnam), L3i (La Rochelle, France), CIL (Athens, Greece), CVC (Barcelona, Spain)
Scientific responsibilities:	LIFAT coordinator (international partnership, digital humanities), Head of the pattern recognition program - CADS Master
Committees, reviewing:	DAS, ICDAR, ICPR, IJDAR, GREC

Summary

Template matching

Template matching is performed by scanning an image I and evaluating the similarity between a template T and an area $W \in I$.



- **Featured-based template matching** extends shape analysis for deformable matching or geometric invariance [1].
- **Correlation-based template matching** extends image comparison for noise robustness and scalability [2].

Template matching

Template matching is a method of parameter estimation.

- The template T is a discrete function $T_{x,y}$ taking values in a window W.
- Template matching chooses position that maximizes the similarity between T and I Eq. (1).
- An application is the L_p-norm with gray-level images Eq. (2).

$$\min_{(i,j)\in I} L_p(i,j) \tag{1}$$

$$L_p(i,j) = \left(\sum_{(x,y)\epsilon W} |I_{x+i,y+j} - T_{x,y}|^p \right)^{\frac{1}{p}} \tag{2}$$

Template matching

The template matching problem is concerned with different parameters.

- $M \times N$, $s \times t$ are the image I and template T sizes, height (M,s) and width (N,t).
- $\mathbb{C}$ is template model search space.
- $O(f)$ the computation cost of the similarity measure f, f could the L_p-norm or other.

$M, N, s, t, \mathbb{C}$ is the search space.
The total computation cost depends of the search space dimension and the $O(f)$ computation.

Template matching

Template matching can be applied to different pattern recognition problems:

	$M \times N$	$s \times t$	$\mathbb{C}$
object detection [3]	large	small	small
image registration [4]	small	large	small
near-duplicate image detection [5]	small	large	large

Summary

Binary template matching

With binary images, binary similarity functions can be applied [6].
They are based on some n_{uv} terms:

n_{11}	the positive matches, i.e. the number of 1 bits that match between y_m and x_m.
n_{00}	the negative matches, i.e. the number of 0 matching bits.
n_{10}, n_{01}	the numbers of bit mismatches.
n	the template / vector size with $n = s \times t = n_{11} + n_{00} + n_{10} + n_{01}$.

Binary template matching

76 ($\times 2$) binary measures can be defined to evaluate either the similarity $S(X, Y)$ or either the dissimilarity $D(X, Y)$ [7].

Measure	$S(X, Y)$	$D(X, Y)$
Sokal and Michener	$\frac{n_{11}+n_{00}}{n}$	$\frac{n_{10}+n_{01}}{n}$
Jaccard and Needham	$\frac{n_{11}}{n_{11}+n_{10}+n_{01}}$	$\frac{n_{10}+n_{01}}{n_{11}+n_{10}+n_{01}}$
Rogers and Tanimoto	$\frac{n_{11}+n_{00}}{n_{11}+n_{00}+2(n_{10}+n_{01})}$	$\frac{2(n_{10}+n_{01})}{n_{11}+n_{00}+2(n_{10}+n_{01})}$
Yule and Kendall	$\frac{n_{11}n_{00}-n_{10}n_{01}}{n_{11}n_{00}+n_{10}n_{01}}$	$\frac{n_{10}n_{01}}{n_{11}n_{00}+n_{10}n_{01}}$

The dissimilarity form of the Sokal and Michener measure $D(X, Y)$ normalizes the L_p-norm in the binary domain.

Binary template matching

The binary measures operate more as measure than as distance.

- Several binary measures are not respecting the Tri-Edge Inequality (TEI) Eq. (3) [8].
- Weighting boosts the classification performances [9].
- A standard weighting value is Eq. (4), to obtain equal weights between foreground / background elements. That is, the commutativity property is not respected $S(X,Y) \neq S(Y,X)$.

$$S(X,Z) \geq S(X,Y) + S(Y,Z) \tag{3}$$

$$\beta = \frac{n_{1x}}{n_{0x}} \qquad \beta \in [0,+\infty[\qquad e.g \quad \frac{\beta n_{11} n_{00} - n_{10} n_{01}}{\beta n_{11} n_{00} + n_{10} n_{01}} \tag{4}$$

Summary

Full-Search methods

Introduction

Full-Search (FS) methods scan the entire image and evaluate the similarity between the pattern and an area [10]. The brute-force method can be optimized with FFT, RLE and bitwise operators for binary similarity measures [11, 5].

Method	**Complexity**	**Constraint**	**Application**
Brute-force	$O(MNst)$	none	little applicable
FFT	$O(M^{*2} \log_2 M^*)$	restricted to the n_{11}, n_{00} matches	when $M \times N$ is large
RLE	$O(kMNst)$ $k << 1$	none	when $M \times N$ is small and $s \times t$ is a constraint
Bitwise	$O(MNk)$ $k = \min(s,t)$	constraining the template size	when $M \times N$ is small and $s \times t$ is not a constraint

M^* is M padded with the template's size and rounded to the above power of two.

Full-Search methods

Bitwise operators

Time processing: FS with the bitwise operators is the fasted method. Modern computers support the comparison of a 4 kB binary word (a 128×256 template) in $0.5 \mu s$.

Image size	**processing time** 128×256 template
1×1	0.5 μs
64×64	2.04 *ms*
128×128	8.19 *ms*
256×256	32.77 *ms*
512×512	131.07 *ms*

Summary

Introduction to template selection

Template definition

object detection [3]	Templates are application-dependent (a character, a logo, . . .).
image registration [4], near-duplicate image detection [5]	Templates characterize salient regions on the images.

Template selection may be posed as follows.

- I is an image of size $M \times N$, $X_k \in I$ is a template of size $s \times t$, we have $\mathtt{C} = (M - s) \times (N - t)$ templates $X_k \in I$.
- Template selection aims to identify X_k, with $X_1, \ldots, X_k, \ldots, X_{\mathtt{C}}$, that best characterizes I for a given pattern recognition problem.

Introduction to template selection

Template selection approaches

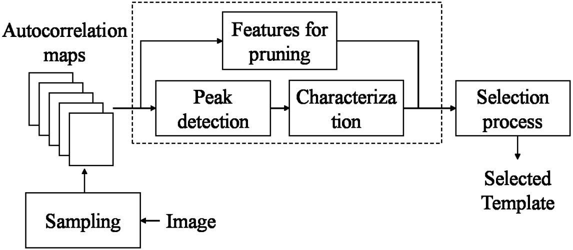
Approach	*Features*	*Characterization*
Shape analysis	Maximum Intensity Variation Number, [12], gradient coherence measure [13]	Fast computation, no reference image, little correlation with matching
Crosscorrelation	Peak sharpness, SNR [4]	High complexity, reference image, near-perfect correlation with matching

Our approach

Approach	*Features*	*Characterization*
Autocorrelation	Peak sharpness, features for pruning, Eccentricity	Quick computation, no reference image, good correlation with matching

Introduction to template selection

The autocorrelation features will look for interesting properties in the autocorrelation domain to accelerate and to make more robust the matching. The baseline process is:

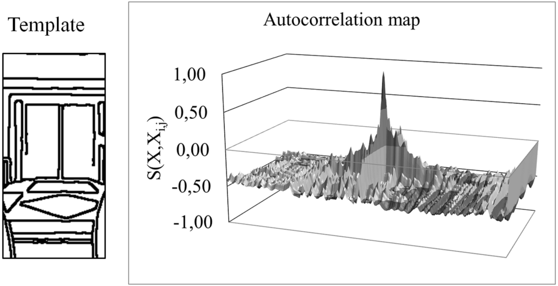


Summary

The autocorrelation map

The autocorrelation map:

- is a matrix W of size $(2s+1) \times (2t+1)$, $i_c = s+1, j_c = t+1$ is the center and locates the peak.
- $W_{i,j}$ provides the similarity measure $S(X, X_{i,j})$ between X and the shifted template $X_{i,j}$ using an offset $\Delta_i = i - i_c, \Delta_j = j - j_c$.
- it is computed from a region of interest of size $(3s+1) \times (3t+1)$.

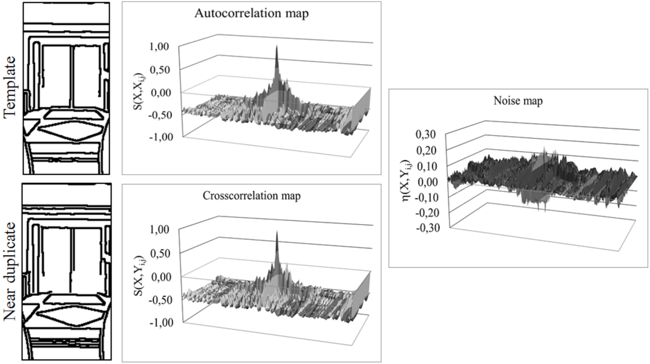


The autocorrelation map

The crosscorrelation / autocorrelation relation is Eq. (5)

$$S(X, Y_{i,j}) = S(X, X_{i,j}) + \eta(X, Y_{i,j}) \qquad \forall i, j \in W \tag{5}$$

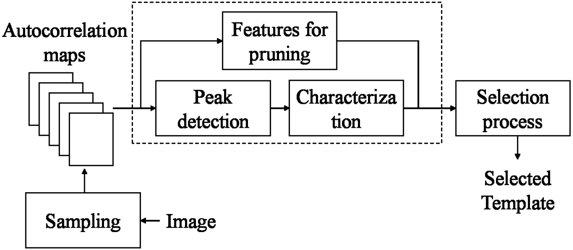
where $S(X, X_{i,j})$, $S(X, Y_{i,j})$ and $\eta(X, Y_{i,j})$ are the autocorrelation, crosscorrelation and noise measures.



Summary

The features for pruning

Introduction



Problem statement: when applying template matching to a significant image size $M \times N$ (e.g. 128×128), the fastest FS method will shift to some tens *ms*.

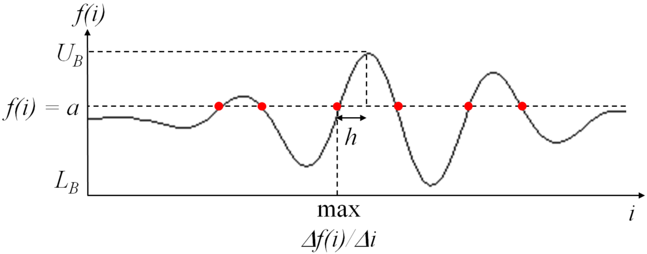
The proposed approach: we index the topology of the autocorrelation maps to tune the matching process to an FS-equivalent mode (i.e. pruning with the FS result guaranty).

The features for pruning

The pruning parameters

Core principle:

- $f(i) = a$ is a constant and provides a $W_{i,j}$ value at i, j.
- $f(i+h)$, with $h \in [0, +\infty[$, is given with $W_{i+h,j} = U_B$.
- we looks for the maximum local derivatives $\Delta f(i)/\Delta i$ $= (f(i+h) - f(i))/h \ \forall i$ with $f(i) = a$.
- that is $W_{i,j}$ is the closest peak element to $W_{i+h,j}$ and h can be used as a pruning parameter.
- a similar process is done to compute $f(j)$.



The features for pruning

The indexing algorithm:

- $S^i = (S^i_1, \ldots, S^i_k, \ldots, S^i_q)$ is an array with a set of q quantified measures $S^i_k \in [L_B, U_B]\ \forall k$.
- at the initialisation, we fix $S^i_k = \emptyset\ \forall k$.
- $\forall i, j$ we obtain the k index with a LUT function $k = LUT\,(W_{i,j})$.
- we set $S^i_k = \min\left(S^i_k, i + h\right)\ \forall i, j$ with $W_{i+h,j} = U_B$, $h \geq 0$.
- a similar process is done for horizontal pruning to get S^j.

The features for pruning

The FS-equivalent algorithm

(*) The autocorrelation map must cover the processed image, then $I \in W$ with $M < 2s + 1$ and $N < 2t + 1$.

(*) B is a boolean matrix of size $M \times N$, we are fixing $B_{i,j} = 0 \; \forall i, j$.

(i) At every pixel location $(i, j) \in B$, if $B_{i,j} = 0$ compute $S(X, Y_{i,j})$.

(ii) Then, get the corresponding k index with the LUT functions and do $B_{i+y,j+x} = 1 \; \forall y \in [0, S^i_k[$ and $\forall x \in [0, S^j_k[$, set $B_{i,j} = 0$.

We compute the acceleration factor as given in Eq. (6)

$$\varpi = (M \times N) / \left(M \times N - \sum_{\forall i,j} B_{i,j} \right) \tag{6}$$

The features for pruning

The noise model

Introduction

- Crosscorrelation differs from autocorrelation due to the noise component $S(X, Y_{i,j}) = S(X, X_{i,j}) + \eta(X, Y_{i,j})$.
- The noise will result in offset values Δ_k when accessing S^i, S^j.
- $\Delta_k > 0$ for additive noise $\eta(X, Y_{i,j}) > 0$.
- $\Delta_k < 0$ for subtractive noise $\eta(X, Y_{i,j}) < 0$.
- To preserve the matching result, the Δ_k offsets should not result in over pruning Eq. (7).

$$S^i_{k+\Delta_k} \overset{not}{>} S^i_k \qquad \forall k \tag{7}$$

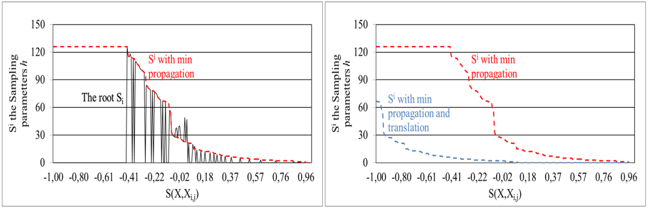
The features for pruning

The noise model

The additive case

- The S^i array appears as a decreasing function as the pruning parameters go down when converging to the peak area.
- We can apply a min propagation $S^i_{k+1} = \min(S^i_k, S^i_{k+1})\ \forall k$ to obtain a monotonically decreasing function.
- We guaranty like this $S^i_{k+\Delta_k} < S^i_k$ with $\Delta_k > 0$ and prevent over pruning with additive noise.
- The process can be extended to S^j.

Transforms to S^i, S^j to guaranty the FS-equivalent mode



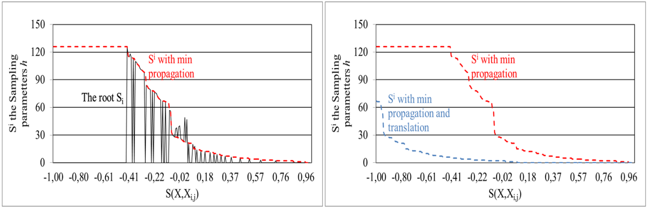
The features for pruning

The noise model

The subtractive case

- We have $\eta(X, Y_{i,j}) < 0$, we fix a threshold $\omega \in [L_B, U_B]$ that guaranties $|\eta(X, Y_{i,j})| < U_B - \omega$.
- We can apply a translation process to S^i with $T_k = LUT(U_B - \omega)$ and $S^i_k = S^i_{k-T_k}\ \forall k$.
- We guaranty like this $S^i_{k+\Delta_k} < S^i_k$ with $\Delta_k > 0$ and prevent over pruning with subtractive noise.
- The process can be extended to S^j.

Transforms to S^i, S^j to guaranty the FS-equivalent mode



The features for pruning

The wavelengths

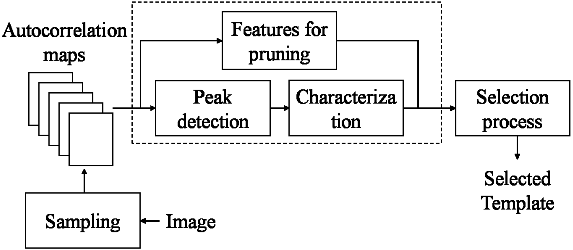
The wavelengths: from the transformed S^i, S^j, the average wavelength for sampling λ is given in Eq. (8). The maximization of this feature characterizes the goodness of the template for pruning.

$$\lambda = \underset{\forall i,j}{\text{mean}}(d(LUT(W_{i,j})) \qquad d(k) = \sqrt{(S^i_k)^2 + (S^j_k)^2} \qquad (8)$$

Summary

Peak detection and characterization

Introduction



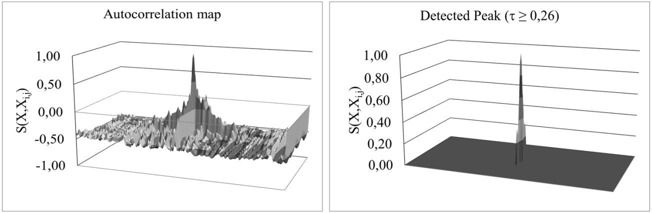
Peak detection and characterization

Peak detection

Problem statement: to characterize the shape of the peak we need to locate it. We can threshold the autocorrelation map with a fix threshold $W_{i,j} > \tau \ \forall i, j$.

Threshold definition:

- τ can be fixed by an expert user [4], that is quite subjective.
- We can determine τ from performance characterization point of view. To not miss peak detection, τ must be fixed to avoid any false negatives *fn*.



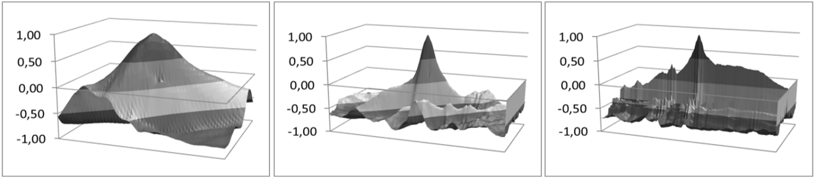
Peak detection and characterization

Peak characterization

Problem statement: once the peak located, we can look for robustness properties when characterizing the peak response.

Features	*Location accuracy*	*Goodness for pruning*	*Robustness*
Sharpness S [4]	maximization	minimization	minimization
Eccentricity E_{CC}	maximization	maximization	maximization

Smooth peak — S "low", E_{CC} "low" | Sharp peak — S "high", E_{CC} "low" | Elongated peak — S "low", E_{CC} "high"



Peak detection and characterization

Peak characterization

Eccentricity E_{cc} is standard image feature, that can be obtained:

- Q is a set $W_{i,j} > \tau \ \forall i, j$.
- $\theta_R \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ describes the direction of the major axis.
- $E_{CC} \in [1, +\infty[$ is eccentricity with $E_{CC} = 1$ a perfect circular disk and $E_{CC} \gg 1$ an elongated region.
- θ_R, E_{CC} are obtained from the central moments μ_{pq} Eq. (9).

$$\mu_{pq} = \sum_{i,j \in Q} (i - i_c)^p (j - j_c)^q \quad \theta_R = \frac{1}{2} \arctan \frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \tag{9}$$

$$E_{CC} = \frac{\mu_{20} + \mu_{02} + \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}}{\mu_{20} + \mu_{02} - \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}}$$

Summary

Performance evaluation

Manga copyright protection

Manga copyright protection is related to a near-duplicate image detection and can be addressed with template matching [5].

- Illegal images are collected from Web portals, at low quality and resolution (e.g. jpg / 128 dpi).
- Legal images are produced per publishers at high resolution and quality.

Performance evaluation

Manga copyright protection

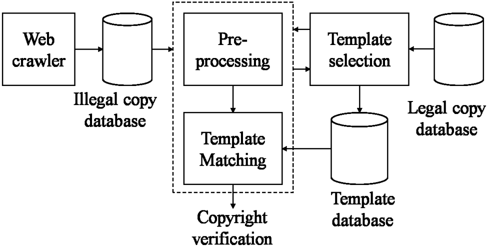
The system proposed in [5].

Web crawler: collects Manga images across the Web and store them into an illegal copy database.

Preprocessing: the line drawing layer is extracted with gray-level conversion and canny-edge detection. The legal images are downsampled for comparison.

Template selection: is applied from legal images.

Template matching: illegal copies are detected with comparison of templates coming from legal pages.



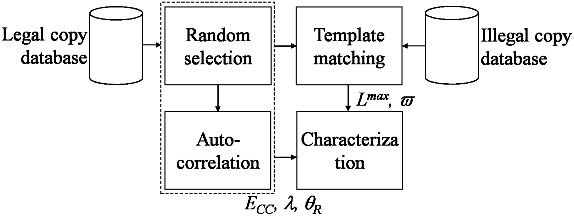
The *MangaOPU* Dataset:

- is composed of $3844 \times 2 = 7688$ legal and illegal image pages (a 3844 class recognition problem).
- is a sample of the *Manga Shukan Shonen Jump* serie[1].
- provides image pages at 128 dpi (a mean page size of 1300×900 pixels).

[1] N° 26, 27, 28, 35, 41, 42, 44, 45, 46 and 48

The characterization protocol:

- Performance characterization is driven in a reference context.
- 30 templates are extracted randomly per page, of size 256×128, and applied for matching.
- The matching is set with the *Yule* measure $S(X,D) \in [-1,1]$ with a weight $\beta = \frac{n_{1x}}{n_{0x}}$, $\tau = 0.26$ and $\omega = 0.12$.
- We keep the template per page with the strongest local maxima L^{max} (samples).
- L^{max}, ϖ are correlated to E_{cc}, λ, θ_R.

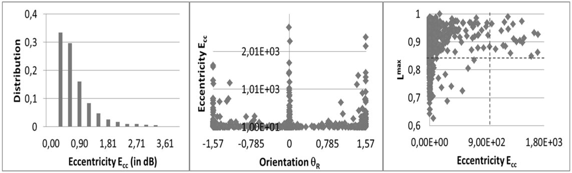


Performance evaluation

Performance characterization

Results:

- $\simeq 95\%$ of peaks are near-blob structures at $E_{CC} < 2$ dB.
- E_{CC} samples are closed to a normal distribution.
- With $E_{CC} \in [0, 2[$ dB, θ_R is little accurate.
- With $E_{CC} > 2$ dB, we have main orientations $|\theta_R| \simeq \{0, \frac{\pi}{2}\}$.
- More a peak converges to a blob structure, more it becomes sensitive to a lowest L^{max}. With $E_{CC} > 2.97$ dB $(\mu + 3\sigma)$, we have $L^{max} \in [0.84, 1]$ with a mean value $L^{max} = 0.92$.

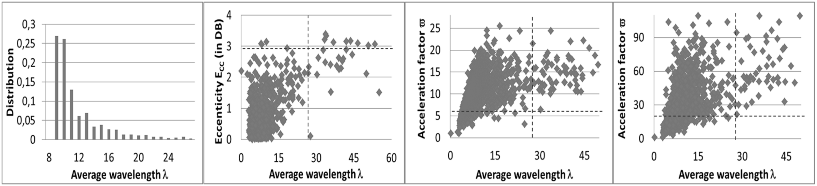


Performance evaluation

Performance characterization

Results:

- λ samples are closed to a normal distribution.
- The λ, E_{CC} maximization are correlated.
- More λ increases, better the acceleration factor ϖ is.
- For intra-class comparisons, with $\lambda > 27.7$ $(\mu + 3\sigma)$ we obtain $\varpi \in [6, 34]$ and a mean value $\varpi = 15.08$.
- For inter-class comparisons, with $\lambda > 27.7$ $(\mu + 3\sigma)$ we obtain $\varpi \in [31, 265]$ and a mean value $\varpi = 90.72$.
- The inter-class case is the major pruning result, the recognition drives $\mathtt{C} - 1$ *tn* comparisons and 1 *tp* comparison.



Performance evaluation

Performance characterization

Results:

- Edge detection requires some tens *ms*.
- The image registration parameters are close to normal distributions, a full coverage is obtained at $-5\sigma, +5\sigma$ with $M \times N = 64 \times 128$.
- It requires then some tens μs to encode and to get the integral image integral from I.
- The FS-equivalent matching operates at the hundred μs level.

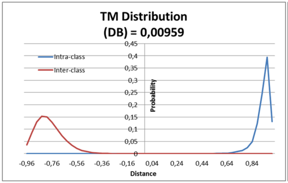
fingerprint	**matching**	
	Encoding / Integral	0.07 *ms*
	FS	4.1 *ms*
	FS with pruning (tp)	0.27 *ms*
	FS with pruning (tn)	0.04 *ms*
35 ms	Total	0.11 / 0.34 ms

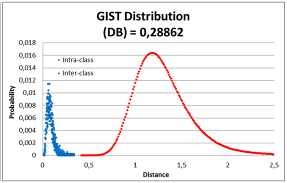
Performance evaluation

Performance characterization

Results:

- We reach separability on the *MangaOPU* dataset (a 3844 class recognition problem).
- The DBI (Davies - Bouldin Index) of the distribution is $DBI = 11,714 \times 10^{-3}$ close from the optimal value $DB = 0$.
- The best near-duplicate descriptor GIST [14] obtains separability with $DBI = 288,627 \times 10^{-3}$ close to the separability upper bound $\frac{1}{3}$.
- GIST is not supposed to preserve separability when faced to a ten or hundred thousands class recognition problem.

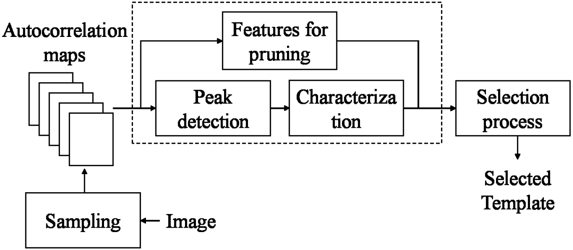




Summary

Sampling and selection rule

Introduction



Sampling and selection rule

Problem statement

- With W of size $\simeq 2s \times 2t = 4n$ ($n = s \times t$), complexity for selection is $O(\complement 4n^2)$ with $\complement \simeq M \times N$.
- $0.5\mu s$ to match a 128×256 template will require 1.87 days of computation for selection with a 1300×1900 image.

Sampling

- The selection doesn't need to be optimal, the important is to detect outlier.
- We sample the image to select the candidate templates by restricting overlapping, to avoid close autocorrelation maps.
- We obtain $\mathcal{C} \ll \complement$ templates such as $(X_1, \ldots, X_k, \ldots, X_{\mathcal{C}}) \in (X_1, \ldots, X_k, \ldots, X_{\complement})$.
- We must set $\mathcal{C}$ large enough in order to reach selection while avoiding unnecessary computation (e.g. $\mathcal{C} \in [5000, 10000]$).

Sampling and selection rule

Selection process

General observations:

- The E_{CC}, λ feature sets are close to normal distributions and their maximisation is correlated.
- The outlier detection with average wavelength λ guaranties a high value range for the acceleration factor ϖ.
- The outlier detection with peak eccentricity E_{CC} guaranties a high value range for the local maxima L^{max}.

Sampling and selection rule

Selection process

The selection rule:

- σ_j, μ_j is the standard deviation, mean of the E_{CC} feature.
- σ_j, μ_j is the standard deviation, mean of the λ feature.
- A standard rule for outlier detection is (a) $E_{CC} > \mu_i + 3\sigma_i$ (b) $\lambda > \mu_j + 3\sigma_j$.
- We select the templates with the rule $a \bullet b$, with $\bullet$ a logical shortcut AND operator.
- As the E_{CC} computation is $\ll$ than λ, with sampling the template selection can be done at the minute scale.

Summary

Conclusions and perspectives

Conclusions

- The binary template matching with selection can be applied to near-duplicate document image detection.
- It is not scale and rotation invariant, but is robust to noise, supports partial skewing and re-sampling.
- It appears as the strongest method of the literature (it is supposed to support recognition problems of some tens to hundreds thousands classes).
- It processes at the hundred μs level for matching and is designed for recognition, not indexing and retrieval.

Short-term:

- To extend experiments for selection and GIST comparison.
- To drive performance evaluation on public dataset (e.g. *Manga*109), this needs degradation models.
- To clarify the τ, ω relation.

Conclusions and perspectives

Perspectives

Mid-term:

- The TEI is not respected, the free-context FS-equivalent methods cannot be applied [15]. Upperbound approximation can be done tacking into account a binary formulation.
- The $\mathbb{C}$ space can be pruned, we propose a reformulation of the Russel-Rao measure through a gaussian registration model (it is segmentation free, it requires template ordering).

Long-term:

- To make the bridge between binary template matching and the detector level . . . .

References I

[1] P. Felzenszwalb, "Representation and detection of deformable shapes," *Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 27, no. 2, pp. 208–228, 2005.

[2] M. Storring and T. Moeslund, "An introduction to template matching," Computer Vision and Media Technology Laboratory (CVMT), Aalborg University, Denmark., Tech. Rep. 01-04, 2001.

[3] S. Mattoccia, F. Tombari, and L. Stefano, "Enhanced low-resolution pruning for fast full-search template matching," in *International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS)*, ser. Lecture Notes in Computer Science (LNCS), vol. 5807, 2009, pp. 109–120.

[4] H. Choi, R. Gupta, and S. Suh, "Quality measurement of template models and automatic template model selection," in *International Conference on Control, Automation and Systems (ICCAS)*, 2012, pp. 1044–1048.

[5] M. Delalandre, M. Iwata, and K. Kise, "Fast and optimal binary template matching, application to manga copyright protection," in *Workshop on Document Analysis Systems (DAS)*, 2014, pp. 298–303.

[6] J. Tubbs, "A note on binary template matching," *Pattern Recognition (PR)*, vol. 22, no. 4, pp. 359–365, 1989.

[7] S. Choi, S. Cha, and C. Tappert, "A survey of binary similarity and distance measures," *Journal of Systemics, Cybernetics and Informatics (SCI)*, vol. 8, no. 1, pp. 43–48, 2010.

[8] B. Zhang and S. N. Srihari, "Discovery of tri-edge inequality with several binary vector dissimilarity measures," in *International Conference on Pattern Recognition (ICPR)*, vol. 4, 2004, pp. 669–672.

[9] S. Cha, C. Tappert, and S. Yoon, "Enhancing binary feature vector similarity measures," *Journal of Pattern Recognition Research (JPRR)*, vol. 1, no. 1, pp. 63–77, 2006.

[10] M. Gharavi-Alkhansari, "A fast globally optimal algorithm for template matching using low-resolution pruning," *Transactions on Image Processing (TIP)*, vol. 10, no. 4, pp. 526–533, 2001.

[11] S. Mukherji, "Fast algorithms for binary cross-correlation," in *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 1, 2005, pp. 340–343.

References II

[12] A. Adnan, “Optimum template selection for image registration using icmm,” in *British Machine Vision Association (BMVC)*, 1998.

[13] A. Bazen and al, “A correlation-based fingerprint verification system,” in *Workshop on Circuits, Systems and Signal Processing (ProRISC)*, 2000, pp. 205–213.

[14] M. Douze and al, “Evaluation of gist descriptors for web-scale image search,” in *International Conference on Image and Video Retrieval (CIVR)*, no. 19, 2009.

[15] W. Ouyang, F. Tombari, S. Mattoccia, and L. D. S. W. Cham, “Performance evaluation of full search equivalent pattern matching algorithms,” *Pattern Analysis and Machine Intelligence (PAMI)*, vol. 34, no. 1, pp. 127–143, 2012.