Pattern Matching & Image Registration

Philippe Latour

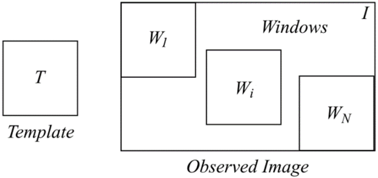
26/11/2014

Table of content

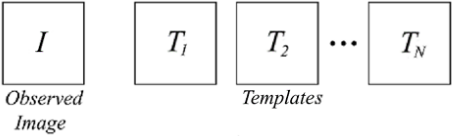
Table of content

Definition: pattern or template matching

- Pattern or template matching is the process of
  - either finding any instance of an image T, called the pattern or the template or the model, within another image I.



  - or finding which ones of the templates $T_1, T_2, \cdots, T_N$ correspond in some way to another image I.



Example: pattern matching

Definition: image registration or alignment

- Image registration is the process of spatially aligning two images of a scene so that corresponding points assume the same coordinates.

Objectives

- **Finding the image transform or warping** that would be needed to fit or align a source image on another one.
- **Counting the number of instances** that matched the pattern.
- **Measuring and assessing** the matching quality.

Components

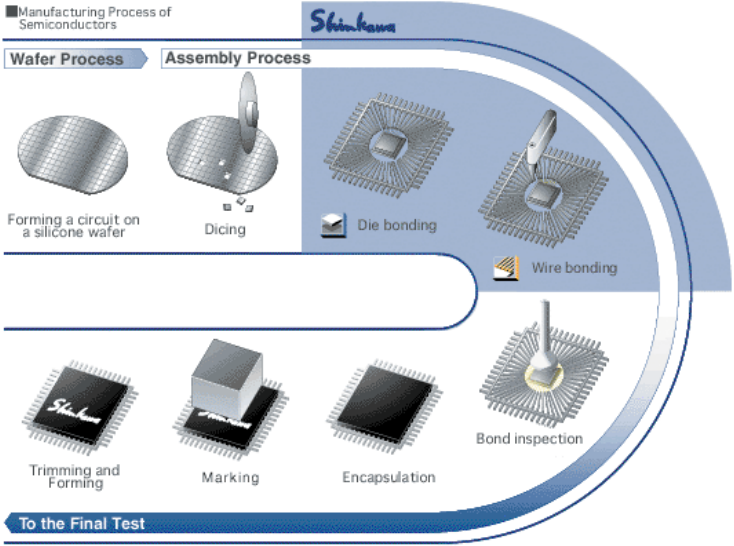
- Find, for some points in the first image, the corresponding point in the second image
  - Either find the correspondence of all pixels of an image
  - Or only find the correspondence of some “interresting” points of an image
- Consider the image variation/transformation/warping between the two images:
  - estimate those which are of interest to our application.
  - specify those to which the system should be insensitive or invariant
- Measure the similarity between the template and its matched instances.

Table of content

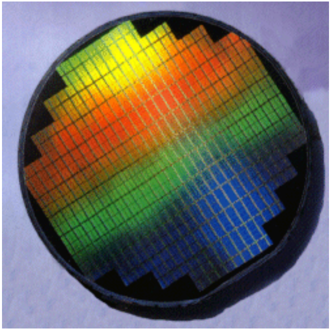
Applications

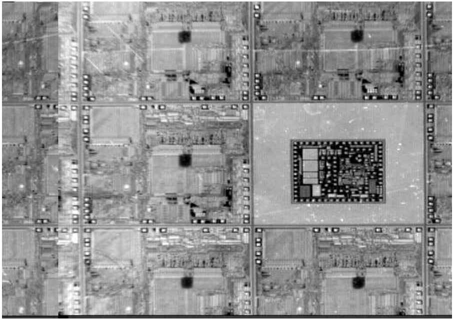
- Stereo and multi-view correspondence:
  - 3D reconstruction
  - Pose estimation
- Panoramic images:
  - Image alignment for stitching
- Machine Vision:
  - Template detection and counting
  - Object alignment -> robot arm control, gauging
  - Model conformance assessment -> NDT, defect detection
- Multi-modalities correspondences:
  - Biomedical images alignment
  - Satellite images fusion
- Robot navigation
- Content Based Image Retrieval (CBIR):
  - Signature

Electronic components manufacturing I

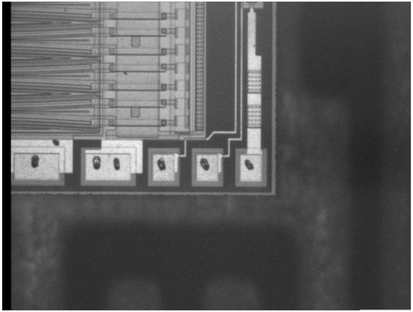


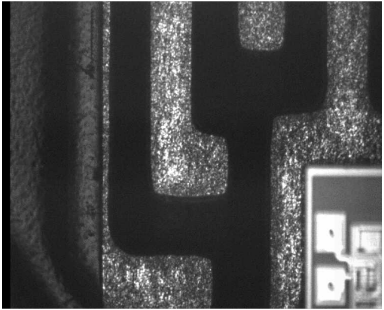
- Wafer dicing

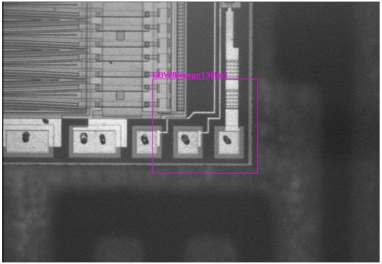


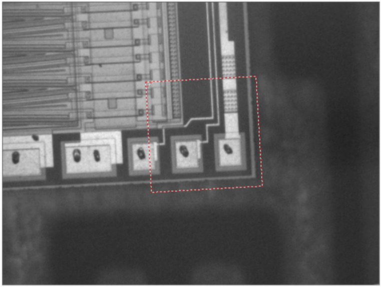


- Die bonding
- Wire bonding





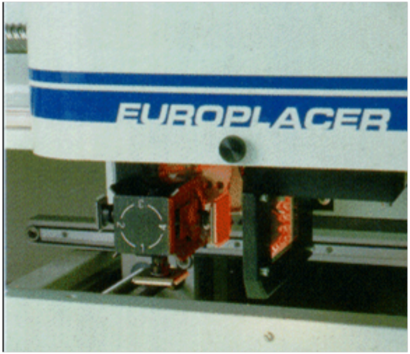

EBW8Image13Roi1

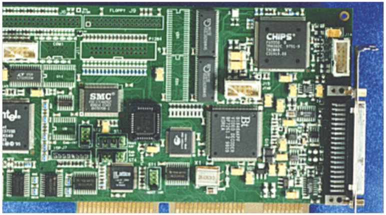


Printed board assembly (pick & place) I

- Position of picked components
- Position of placement area
- Control of welding after the process

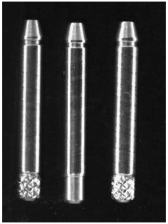
Printed board assembly (pick & place) II

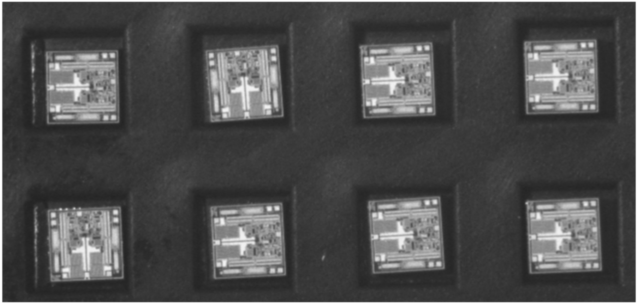




Pattern matching inspection I

- Control of presence/absence
- Control of position and orientation
- Control of the component type





Pattern matching inspection II

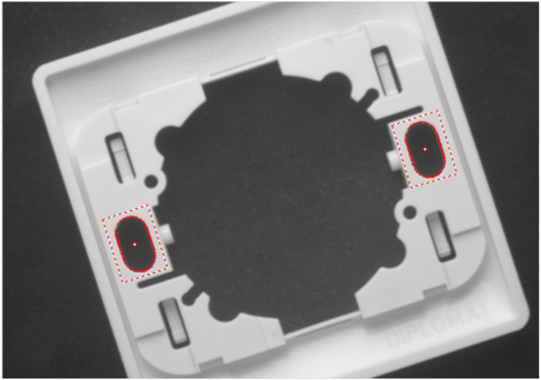
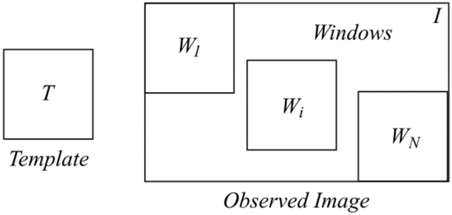


Table of content

Naive solution - pattern pixels distance I

- Define, inside the observed image I, all the windows W_i of the same size (width w_T x height h_T) as the template T. If $(x, y) = (k_i, l_i)$ is the center of W_i, then $W_i\left(x, y\right) = I\left(x + k_i - \frac{w_T}{2}, y + l_i - \frac{h_T}{2}\right)$
- For each window W_i, compute the euclidian distance between T and W_i:

$$dist\left(T, W_i\right) = \sum_{u=0}^{w_T - 1}\sum_{v=0}^{h_T - 1}\left[T\left(u, v\right) - W_i\left(u, v\right)\right]^2 \tag{1}$$

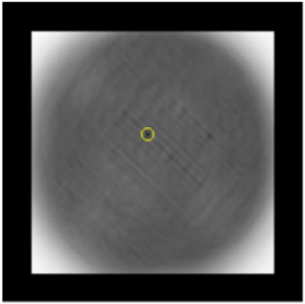


Naive solution - pattern pixels distance II

- Create a distance map that contains for each position W_i the computed distance to T

$$D(k,l) = \begin{cases} dist\left(T, W_{i(k,l)}\right) & when \quad \begin{cases} \frac{w_T}{2} \leq k < w_I - \frac{w_T}{2} \\ \frac{h_T}{2} \leq l < h_I - \frac{h_T}{2} \end{cases} \\ 0 & otherwise \end{cases} \tag{2}$$

- Find the position of the minimum in these map

Pixel-based approach I

- The approach is to shift or warp the images relative to each other and to look at how much the pixels agree
- A suitable similarity or dissimilarity measure must first be chosen to compare the images
- The similarity or dissimilarity measure depend on the image characteristics to which it is necessary to be invariant.
  - Lighting conditions (linear gain and offset)
  - Noise
  - "Small" rotation or scaling
  - Thinning
  - -> Define the similarity/dissimilarity measure

Pixel-based approach II

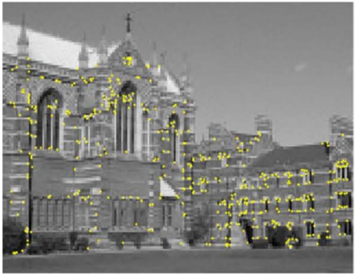
- Then, we need to decide which kind of warping is eligible!
  - Translation, rotation, scaling, affine transform
  - -> Define the search space
- The search space is the parameter space for the eligible warping (the set of all the parameters giving rise to an eligible transformation).
  - Translation -> 2D search space
  - Translation + rotation -> 3D search space
  - Translation + rotation + isotropic scaling -> 4D search space
  - Affine transform -> 6D search space
  - Projective transform -> 8D search space
- Finally, the search technique must be devised
  - Trying all possible alignment (a full search) is often impractical!
  - So hierarchical coarse-to-fine techniques based on image pyramids are often used.

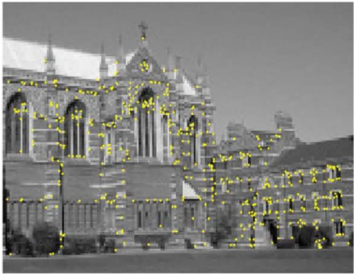
Other solution: feature point correspondences I

There are mainly two paradigms to compare two images:

- Pixel based:
  - Compare all pairs of corresponding (=located at the same place in the image, possibly after warping one image) pixels
  - Then compute a global score based on the individuals comparisons
- Feature based:
  - Find "informatives" feature points in each images
  - Then associate each feature point of one image to a feature point of the other image
  - Compute the transformation/warping that enable the feature point in the left image to fit their corresponding point in the rigth image

Other solution: feature point correspondences II





Feature-based approach

- Feature points have also been referred to as critical points, interest points, key points, extremal points, anchor points, landmarks, control points, tie points, corners, vertices, and junctions in the literature.
- We need to decide what is a feature point
  - Corners, junctions, edges, blob center, ...
  - Compute a cornerness function and suppress non-maxima
  - Design to be invariant to some image variation
- Then we have to characterize and describe them (position, image gradient or moment, cornerness, ...) to find the best correspondance between feature points in each images
- We need to decide which kind of warping is admissible!
  - How to find the best correspondences
  - Robust methods (Ransac, ICP)

Pixel-based versus feature based approach I

	Pixel based	Feature based
Pattern information usage	Use all pixels in the pattern in an uniform way. -> No need to analyze or understand the pattern.	Find and use pattern features (most informative part of the pattern). -> Sensitive operation.
Occlusion or pose variation	Sensitive	Could be design to be insensitive
Sub-pixel accuracy	Interpolation of the similarity/dissimilarity measure	Naturally accurate at the sub-pixel level.
Admissible warping	The choice has to be done at the beginning of the process (orientation and scaling)	Mostly insensitive to differences in orientation and scaling
Noise and lighting conditions	Sensitive	Naturally much more insensitive

Pixel-based versus feature based approach II

	Pixel based	Feature based
Rigid pattern warping	Mostly limited to rigid pattern warping	Enable non-rigid warping.
Dimensionality of the search space	Mostly limited to low dimensionality (the search time is exponential in the search space dimensionality)	Higher dimensionality search space are more easily reachable
Implementation	Easy to implement, natural implementation on GPUs	Much more difficult to implement and/or to optimize
Complexity	Complexity proportionnal to the image size. Need specific search strategies to reach real-time.	Complexity roughly proportionnal to the number of feature points (depend more on the content of the scene than on the image size)

Table of content

Pixel-based approach: Similarity/dissimilarity measures

- Given two sequences of measurement
  - $X = \{x_i \mid i = 1, \cdots, n\}$
  - $Y = \{y_i \mid i = 1, \cdots, n\}$
  - X and Y can represent measurements from two objects or phenomena. Here, in our case, we assume they represent images and x_i and y_i are the intensities of the corresponding pixels in the images.
- The similarity (dissimilarity) between them is a measure that quantifies the dependency (independency) between the sequences.

Pixel-based approach: Similarity/dissimilarity measures

Similarity / Correlation / Score	Dissimilarity / Distance
Pearson correlation coefficient Tanimoto Measure Stochastic Sign Change Deterministic Sign Change Minimum Ratio Spearman's Rho Kendall's Tau Greatest Deviation Ordinal Measure Correlation Ratio Energy of Joint Probability Distribution Material Similarity Shannon Mutual Information Rényi Mutual Information Tsallis Mutual Information F-Information Measures	L1 Norm Median of Absolute Differences Square L2 Norm Median of Square Differences Normalized Square L2 Norm Incremental Sign Distance Intensity-Ratio Variance Intensity-Mapping-Ratio Variance Rank Distance Joint Entropy Exclusive F–Information

Pixel-based approach: Similarity/dissimilarity measures

Similarity / Correlation / Score	Dissimilarity / Distance
Pearson correlation coefficient	*L1 Norm*
Tanimoto Measure	*Median of Absolute Differences*
Stochastic Sign Change	Square L2 Norm
Deterministic Sign Change	*Median of Square Differences*
Minimum Ratio	*Normalized Square L2 Norm*
Spearman's Rho	*Incremental Sign Distance*
Kendall's Tau	*Intensity-Ratio Variance*
Greatest Deviation	*Intensity-Mapping-Ratio Variance*
Ordinal Measure	*Rank Distance*
Correlation Ratio	*Joint Entropy*
Energy of Joint Probability Distribution Material	*Exclusive F–Information*
Similarity Shannon Mutual Information	
Rényi Mutual Information	
Tsallis Mutual Information	
F-Information Measures	

Pearson correlation coefficient

The correlation coefficient between sequences $X = \{x_i \mid i = 1, \cdots, n\}$ and $Y = \{y_i \mid i = 1, \cdots, n\}$ is given by

$$r = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3}$$

where

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad \text{and} \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

which can also be written as

$$r = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{(x_i - \bar{x})}{\sigma_x}\right)\left(\frac{(y_i - \bar{y})}{\sigma_y}\right) \tag{4}$$

or

$$r = \frac{1}{n}\bar{X}^t\,\bar{Y} \tag{5}$$

Pearson correlation coefficient - FFT

- To speed-up the search step when the score is, in some way, related to a correlation coefficient, we can use FFT algorithm:
  - Y represents the 2-D image inside which a 2-D template is to be found
  - X represents the template padded with zeros to be the same size as Y
  - The best-matching template window in the image is located at the peak of

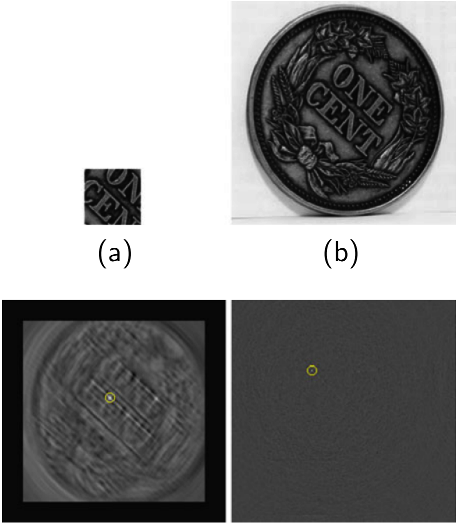
$$C\left[X, Y\right] = \mathscr{F}^{-1}\left\{\mathscr{F}\left\{X\right\}\mathscr{F}^{*}\left\{Y\right\}\right\} \tag{6}$$

- Phase correlation: the information about the displacement of one image with respect to another is included in the phase component of the cross-power spectrum of the images:

$$C_p\left[X, Y\right] = \mathscr{F}^{-1}\left\{\frac{\mathscr{F}\left\{X\right\}\mathscr{F}^{*}\left\{Y\right\}}{\left\|\mathscr{F}\left\{X\right\}\mathscr{F}^{*}\left\{Y\right\}\right\|}\right\} \tag{7}$$

Pearson correlation coefficient map

(a) A template X.
(b) An image Y containing the template X.
(c) The correlation image $C[X, Y]$ with intensity at a pixel showing the correlation coefficient between the template and the window centered at the pixel in the image.
(d) The real part of image $C_p[X, Y]$, showing the phase correlation result with the location of the spike encircled.



(a) (b)

(c) (d)

Spearman rank correlation or Spearman's rho I

- The Spearman rank correlation or Spearman's Rho (ρ) between sequences $X = \{x_i \mid i = 1, \cdots, n\}$ and $Y = \{y_i \mid i = 1, \cdots, n\}$ is given by

$$\rho = 1 - \frac{6 \sum_{i=1}^{n} \left[R(x_i) - R(y_i)\right]^2}{n(n^2 - 1)} \tag{8}$$

  where $R(x_i)$ and $R(y_i)$ represent ranks of x_i and y_i in images X and Y

- Remark: To eliminate possible ties among discrete intensities in images, the images are smoothed with a Gaussian of a small standard deviation, such as 1 pixel, to produce unique floating-point intensities.

Spearman rank correlation or Spearman's rho II

- Comparison with the Pearson correlation coefficient:
  - ρ is less sensitive to outliers and, thus, less sensitive to impulse noise and occlusion.
  - ρ is less sensitive to nonlinear intensity difference between images than Pearson correlation coefficient.
  - Spearman's ρ consistently produced a higher discrimination power than Pearson correlation coefficient.
  - Computationally, ρ is much slower than r primarily due to the need for ordering intensities in X and Y

- If x_i and y_i, for $i = 0, ..., n$, show intensities of corresponding pixels in X and Y , then for $i \neq j$, two possibilities exist:
  - Either concordance : $sign(x_j - x_i) = sign(y_j - y_i)$
  - Or discordance : $sign(x_j - x_i) = -sign(y_j - y_i)$
- Assuming that out of possible C_n^2 combinations, N_c pairs are concordants and N_d pairs are discordants, Kendall's τ is defined by:

$$\tau = \frac{N_c - N_d}{n\left(n-1\right)/2} \tag{9}$$

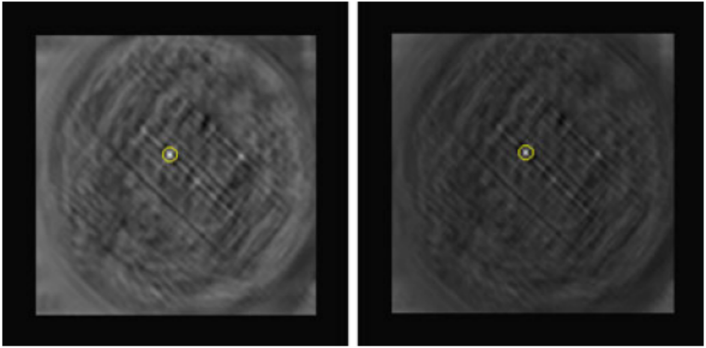
- If bivariate (X, Y) is normally distributed, Kendall's τ is related to Pearson correlation coefficient r by:

$$r = \sin\left(\pi\tau/2\right) \tag{10}$$

Kendall's tau II

- Comparison with other similarity measures:
  - Pearson correlation coefficient can more finely distinguish images that represent different scenes than Kendall's τ
  - Conversely, Kendall's τ can more finely distinguish similar images from each other when compared to Pearson correlation coefficient
  - Spearman's ρ and Kendall's τ have the same discrimination power when comparing images of different scenes
  - Kendall's τ is one of the costliest similarity measures

Spearman's rho and Kendall's tau maps



(a) (b)

(a) Spearman's Rho
(b) Kendall's Tau.

Feature point detection

Feature points in an image carry critical information about scene structure
-$>$ They are widely used in image analysis.

In image registration, knowledge about corresponding points in two images is required to spatially align the images.

It is important that detected points be independent of noise, blurring, contrast, and geometric changes
-$>$ the same points can be obtained in images of the same scene taken under different environmental conditions and sensor parameters.

A large number of point detectors have been developed throughout the years ...

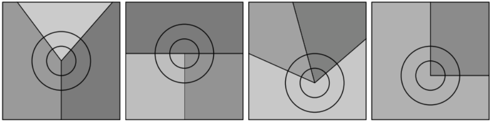
Feature point category

- Correlation-based detectors
- Edge-based detectors
- Model-based detectors
- Uniqueness-based detectors
- Curvature-based detectors
- Laplacian-based detectors
- Gradient-based detectors
- Hough Transform-based detectors
- Symmetry-based detectors
- Filtering-based detectors
- Transform Domain detectors
- Pattern Recognition-based detectors
- Moment-based detectors
- Entropy-based detectors

Feature point category

- Correlation-based detectors
- Edge-based detectors
- Model-based detectors
- Uniqueness-based detectors
- Curvature-based detectors
- Laplacian-based detectors
- Gradient-based detectors
- Hough Transform-based detectors
- Symmetry-based detectors
- Filtering-based detectors
- Transform Domain detectors
- Pattern Recognition-based detectors
- Moment-based detectors
- Entropy-based detectors

Correlation-based detectors I

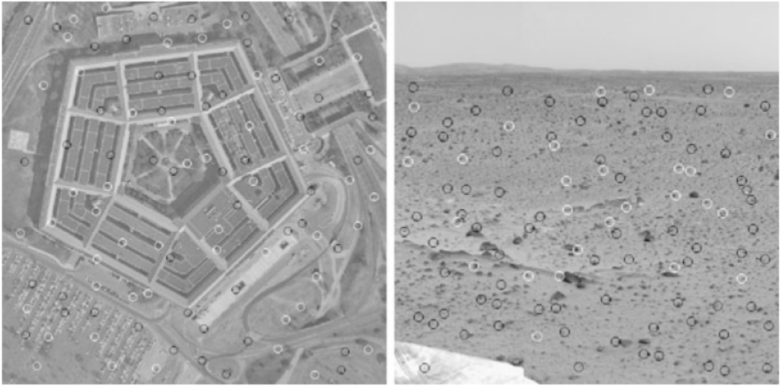


- The angle between the line connecting pixel (x, y) to the ith pixel on the smallest circle and the x-axis is θ_i, and the intensity at the ith pixel is $I_1(\theta_i)$
- If $\bar{I}_j(\theta_i)$ represents the normalized intensity at θ_i in the jth circle, then

$$C(x, y) = \sum_{i=1}^{n} \prod_{j=1}^{m} \bar{I}_j(\theta_i) \tag{11}$$

  is used to measure the strength of a vertex or a junction at (x, y) .
- Pixel (x, y) is then considered a **corner** if $C(x, y)$ is locally maximum.

Correlation-based detectors II



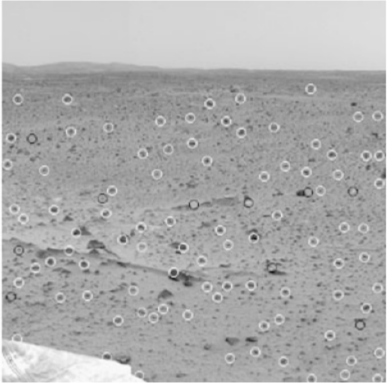
(a) (b)

Laplacian-based detectors I

- A number of detectors use either the Laplacian of Gaussian (LoG) or the difference of Gaussians (DoG) to detect points in an image.
  - The DoG operator is an approximation to the LoG operator
  - The best approximation to the LoG operator of standard deviation σ is the difference of Gaussians of standard deviations σ and 1.6σ. That is $\nabla^2 G(\sigma) = \frac{1.6[G(1.6\sigma) - G(\sigma)]}{\sigma^2}$
- Local extrema of LoG or its approximation DoG detect **centers of bright or dark blobs** in an image.
  - So, they are not as much influenced by noise as points representing corners and junctions and points detected by the LoG operator are generally more resistant to noise than points detected by vertex and junction detectors.
- SIFT (Scale Invariant Feature Transform) used the difference of Gaussians (DoG) to find points in an image.

Laplacian-based detectors II


(a)


(b)

Pattern and image warping or transform

- Pattern or Image warping
  - The eligible warping defines the search space
    - Translation
    - Rotation
    - Isotropic / anisotropic scaling
    - Affine / projective transform
    - Non-linear warping
  - The insensitivity properties guide the choice of a score/distance measure and the choice of a feature point detector
    - Noise
    - Lighting conditions
    - "Small" rotations or scaling
    - Template thinning
- Applying the warping
  - to the pattern or the image or both ?
  - or to the feature points ?
- Image resampling and sub-pixel accuracy ?

Transform parameter space search strategy

- Similarity/Dissimilarity measure optimization
  - Full search
  - Steepest descent
  - Conjugate gradient
  - Quasi-Newton method
  - Levenberg-Marquardt
  - Simulated annealing
- Transform computation from feature points correspondence
  - Ransac
  - ICP (Iterative Closest Point)

Table of content

- Introduction
- Applications
- Some solutions and their corresponding approach
- Pattern matching components
- Implementation speed-up
  - FFT
  - Multiresolution: Coarse to fine
  - Hybrid approach: feature extraction in one image only

Multiresolution - Coarse-to-fine approach

- Compute image and pattern down-scaled pyramids.
- Proceed to a full search of the most reduced (coarser) pattern within the most reduced image.
- Find a number of eventual candidates at the coarsest scale by a full search.
- For each candidates at a given scale:
  - Upscale the image and the candidate and look for the best matching pattern location in a neighbourhood of the candidate.
  - Reduce the number of candidates
  - If the finer scale has not yet been reached, proceed to the next scale level

Hybrid approach: feature extraction in one image only

- Search for some feature points in the pattern
- Scan the transform parameter space following a given strategy:
  - Transform the feature points following the current eligible warping parameters
  - Superimpose the transformed pattern feature points on the reference image
  - At each pattern feature points location in the reference image, check if a compatible point exists in the reference image and measure its similarity/dissimilarity score.
  - Compute a global measure of similarity/dissimilarity by adding all the individual scores.
  - Find the optimum of this measure on the search space.

Bibliography

A. Goshtasby.
Image registration – Principles, Tools an Methods.
Springer-Verlag London, 2012, DOI: 10.1007/978-1-4471-2458-0

R. Brunelli.
Template matching techniques in computer vision – Theory and practice. John Wiley & Sons, 2009.

R. Szeliski.
Image alignment and stitching: a tutorial. Technical Report MSR-TR-2004-92, Microsoft Research, 2006