Visual Feature Spaces for Image Representation and Processing: *The Multiscale Local Jet*

Antoine Manzanera
ENSTA-ParisTech

Workshop AMINA 2010 - Oct, 19, 2010

Objectives and problem statement

Our global objective is to propose a unified framework for representing and processing the visual data, trying to reach the maximal:

- *Universality*: from the lowest (denoising, registration,...) to the highest level (recognition, understanding,...) of vision.
- *Computational efficiency*: tractable complexity for embedded video systems.

Objectives and problem statement

The principle of our framework is to adjoin to the video data an alternate (feature space) data structure whose metrics is related to visual similarity, and made up of different components:

- Feature space (local jet) projection.
- Quantisation (Codebook) of the feature space.
- Nearest Neighbour Search in the feature space.
- Indexing of the feature space for back-projection.

Outline

1. Introduction
   - Objectives and problem statement
   - Related works
2. Local jet features
   - Local jet based similarity
   - LJ feature space
   - Metrics and invariance
   - Computation and data structures
3. Image processing
   - NL-means filtering
   - Optical flow
   - Background subtraction
4. Image representation
   - Codebook histograms
   - Singularities and Modes in the feature space

Related works

Manifold Image Processing

The projected data form a manifold in some feature space [Peyré 09]. Image processing is made by transformation of the manifold, followed by a back-projection onto the image space.

Filter Banks and Codebook

Many visual representation frameworks, for texture (e.g. textons) or objects (e.g. visual bag of features) are based on filter banks and clustering [Freeman 91], [Rubner 99].

Nearest Neighbour Search

We use Binary Search Trees to represent highly sparse sets from the feature space with a minimal amount of memory, and efficiently perform Nearest Neighbour Search [Arya and Mount 07].

Outline

Local derivatives and visual similarity

SSD based similarity

$$d_f(\mathbf{x},\mathbf{y}) = \sum_{\mathbf{i}\in\mathcal{W}(\mathbf{O})} k(||\mathbf{i}||)(f(\mathbf{x}+\mathbf{i}) - f(\mathbf{y}+\mathbf{i}))^2$$

Taylor expansion

$$f(\mathbf{x}+\mathbf{c}) = \sum_{k=0}^{r}\sum_{i=0}^{k}\binom{k}{i} c_1^{k-i} c_2^{i} \frac{\partial^k f}{\partial x_1^{k-i}\partial x_2^{i}}(\mathbf{x}) + o(||\mathbf{c}||^r)$$

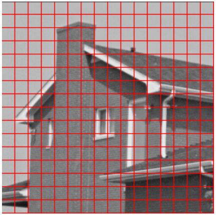
Local jet based similarity

$$d_f(\mathbf{x},\mathbf{y}) \simeq \sum_{i+j\leq r} \alpha_{(i,j)}(f_{ij}(\mathbf{x}) - f_{ij}(\mathbf{y}))^2, \text{ with } f_{ij} = \frac{\partial^{i+j} f}{\partial x_1^i \partial x_2^j}$$

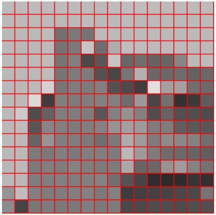
Local jet based representation

The Local jet seen as a summary of the patch information (Reconstruction of the patch using Taylor expansion)...

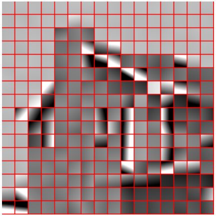
1. Original 15×15 patches
2. Order 0 (1d feature)
3. Order 1 (3d feature)
4. Order 2 (6d feature)



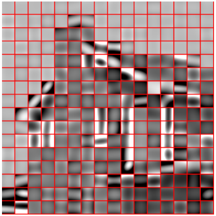
(1)



(2)



(3)



(4)

Gaussian multiscale local jet

We do not use patches, but Gaussian Local Jet at a certain scale:

Projection in the Local Jet space

$$\mathbf{x} \mapsto \hat{\mathbf{x}} = (a_{ij}^{\sigma} f_{ij}^{\sigma}(\mathbf{x}))_{i+j\leq r}$$

Gaussian multiscale local jet

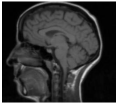
$$f_{ij}^{\sigma} = f \star \frac{\partial^{i+j} G_{\sigma}}{\partial x_1^i \partial x_2^j}$$

Normalised local jet

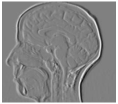
$$a_{ij}^{\sigma} = \frac{\sigma^{i+j}}{i+j+1}$$

- G_{σ} is the 2d Gaussian function with variance σ^2.
- σ^{i+j} is the scale space normalisation [Lindeberg 98].
- $i+j+1$ is the number of $(i+j)^{\text{th}}$ order derivatives.

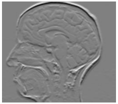
Multiscale "Cartesian" Local jet



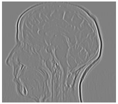
$f_{00}^{1.0}$



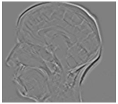
$f_{10}^{1.0}$



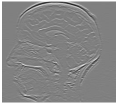
$f_{01}^{1.0}$



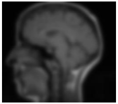
$f_{20}^{1.0}$



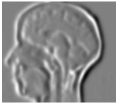
$f_{11}^{1.0}$



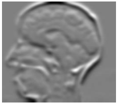
$f_{02}^{1.0}$



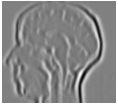
$f_{00}^{4.0}$



$f_{10}^{4.0}$

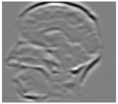


$f_{01}^{4.0}$



$f_{20}^{4.0}$

$f_{11}^{4.0}$

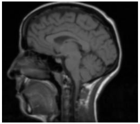


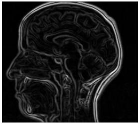
$f_{02}^{4.0}$

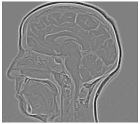
Rotation invariance

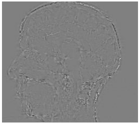
$$
\begin{aligned}
f &= f_{00} \\
f_g &= (f_{10}^2 + f_{01}^2)^{1/2} \\
f_t &= 0 \\
f_{gg} &= (f_{20} f_{10}^2 + 2 f_{11} f_{10} f_{01} + f_{02} f_{01}^2) f_g^{-2} \\
f_{tt} &= (f_{20} f_{01}^2 - 2 f_{11} f_{10} f_{01} + f_{02} f_{10}^2) f_g^{-2} \\
f_{gt} &= (f_{10} f_{01} (f_{20} - f_{02}) + f_{11} (f_{01}^2 - f_{10}^2)) f_g^{-2}
\end{aligned}
$$

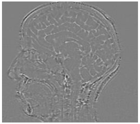
Multiscale Rotation Invariant Local jet

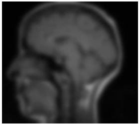

$f^{1.0}$

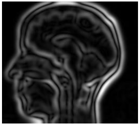

$f_g^{1.0}$

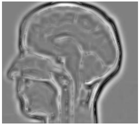

$f_{gg}^{1.0}$

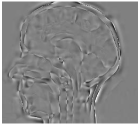

$f_{gt}^{1.0}$

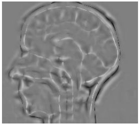

$f_{tt}^{1.0}$


$f^{4.0}$


$f_g^{4.0}$


$f_{gg}^{4.0}$


$f_{gt}^{4.0}$


$f_{tt}^{4.0}$

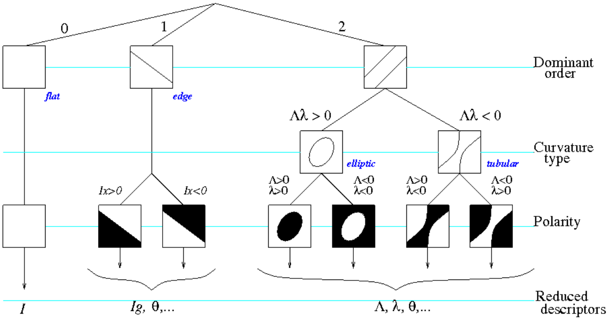
Contrast change (and inversion !) Invariant Local jet

Gradient direction

Main absolute curvature direction

Local jet pixel categories and similarity metrics

A reduced description useful for pixel comparison can be obtained by categorizing the local jet. See also [Crozier 10].



Some metrics

Single scale distance

$$d_f^\sigma(x,y) = \sum_{i+j\leq r} (f_{ij}^\sigma(x) - f_{ij}^\sigma(y))^2$$

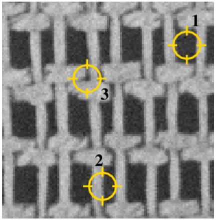
Pan-scalic distance

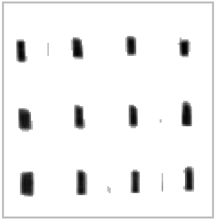
$$D_f^S(x,y) = \sum_{i+j\leq r, \sigma\in S} (f_{ij}^\sigma(x) - f_{ij}^\sigma(y))^2$$

Trans-scalic distance

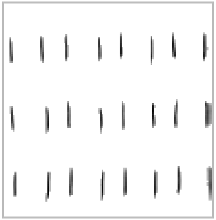
$$d_f^S(x,y) = \min_{(\sigma_1,\sigma_2)\in S^2} \sum_{i+j\leq r} (f_{ij}^{\sigma_1}(x) - f_{ij}^{\sigma_2}(y))^2$$

Local jet similarity map: one example

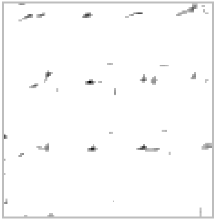




(1)

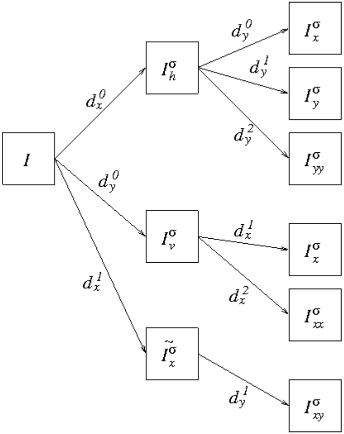


(2)



(3)

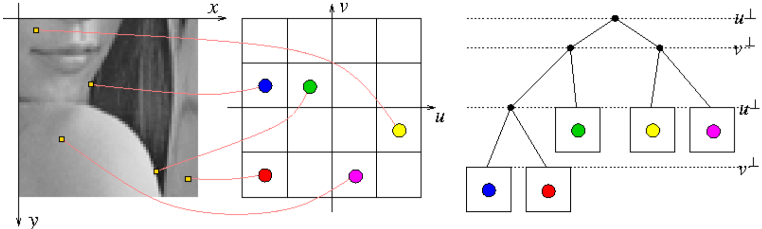
Parallel computation of the Local Jet



The local jet features lend themselves to a high level of parallel / cascaded computation. As an example, the recursive computation [Van Vliet 98] of the local jet at order 2 requires 9 couples of image scans, with only 2 levels of dependance.

Data projection and representation

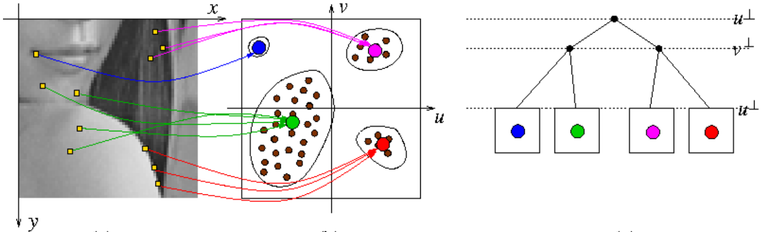
- The pixel data are projected in the feature space.
- The feature vectors are collected within a binary search tree.
- Every feature vector is attached to a pixel index.



(a) (b) (c)

Quantisation of the feature space

- The feature space can be quantised through vector quantisation.
- Every codebook word keeps record of a set of pixel indices.



(a) (b) (c)

Outline

NL-means in the feature space

The NL-means filter calculates a weighted average of every pixel, with the weights defined as a function of local similarity. Here we simply use the distance in the feature space...

$$\omega(u,v) = e^{-\frac{||u-v||^2}{h^2}}$$

with:

$||.||$ a norm in the feature space.

h a decay parameter.

...and perform the average on a neighbourhood of x in the image space (Limited Range),

or on a neighbourhood of $\hat{x}$ in the feature space (Unlimited range).

Limited Range LJ-NL-Means

The weights (in the LJ space) are calculated in a limited neighbourhood of $\mathbf{x}$ *in the image space*:

Limited range NL-means

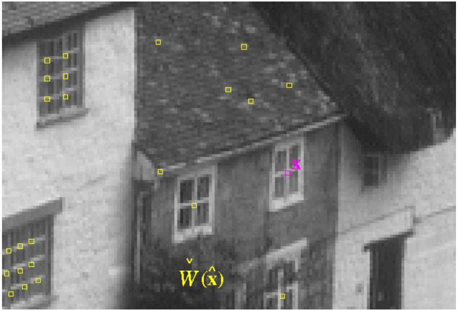
$$f_{LR}^{NL}(x) = \frac{1}{\zeta(\mathbf{x})} \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} f(y)\omega(\hat{\mathbf{x}}, \hat{\mathbf{y}})$$

Unlimited Range LJ-NL-Means

The weights (in the LJ space) are calculated in a limited neighbourhood of $\hat{\mathbf{x}}$ *in the LJ space*:

Unlimited range NL-means

$$f_{UR}^{NL}(\mathbf{x}) = \frac{1}{\xi(\mathbf{x})} \sum_{u \in \mathcal{W}(\hat{\mathbf{x}})} f(\check{\mathbf{u}})\omega(\hat{\mathbf{x}}, \mathbf{u})$$



LJ-NL-means: Relative computation time

Patch based	LR Local Jet
100	16.1

LR LJ ($|\mathcal{N}(x)| = 17 \times 17$), order 2.

1 scale	2 scales	3 scales	4 scales
14.0	36.4	73.5	93.3

UR LJ ($|\mathcal{W}(\hat{x}_f)| = 30$), order 2, exact search

$\varepsilon = 0.0$	$\varepsilon = 1.0$	$\varepsilon = 3.0$	$\varepsilon = 10.0$
93.3	35.1	18.8	10.4

Same (4 scales), approximate search

No quantisation	1622 words	795 words
18.8	45.6 (40.9)	23.0 (19.7)

Same ($\varepsilon = 3.0$), with quantisation (In parentheses: quantisation time).

LJ-NL-means: results

1. Classical Patch based ($|\mathcal{N}(x)| = 17 \times 17$)
2. Limited range ($|\mathcal{N}(x)| = 17 \times 17$)
3. Unlimited range, exact search ($|\mathcal{W}(\hat{x}_f)| = 30$, single scale).
4. Same, 4 scales.
5. Same, approximate search ($\varepsilon = 3.0$).
6. Same, with quantisation (≈ 800 word codebook).


(1)


(2)


(3)

Apparent motion in the feature space

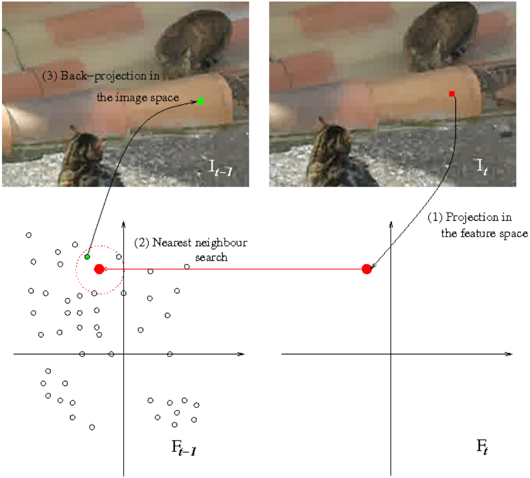
The optical flow estimation can be expressed by a simple nearest neighbour search in the feature space:

$$u(f_{t-1}, f_t, x) = \arg \min_{v \in \mathcal{F}_{f_{t-1}}} d^F(\hat{x}_{f_t}, v)$$

followed (in the case of a quantised feature space) by an optimization in the image space:

$$y(f_{t-1}, f_t, x) = \arg \min_{z \in \mathcal{F}^{-1}_{f_{t-1}}(u(f_{t-1}, f_t, x))} d^I(x, z)$$

Apparent motion in the feature space



Feature space optical flow: NL-Mean formulation

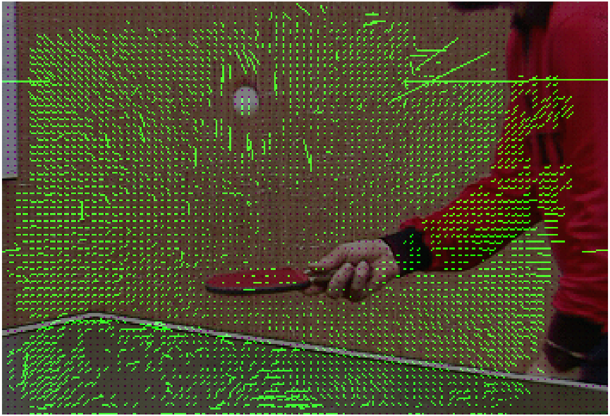
A more general formulation can be obtained following the NL-Mean framework:

$$\mathbf{y}(f_{t-1}, f_t, \mathbf{x}) = \frac{1}{\xi^F(\mathbf{x})} \sum_{k=1}^{m} \left(\frac{\omega^F(\hat{\mathbf{x}}_{f_t}, \nu_k^{\mathcal{F}_{f_{t-1}}}(\hat{\mathbf{x}}_{f_t}))}{\xi^I(\mathbf{x})} \sum_{\mathbf{z} \in \mathcal{F}_{f_{t-1}}^{-1}(\nu_k^{\mathcal{F}_{f_{t-1}}}(\hat{\mathbf{x}}_{f_t}))} \omega^I(\mathbf{x}, \mathbf{z})\mathbf{z} \right)$$

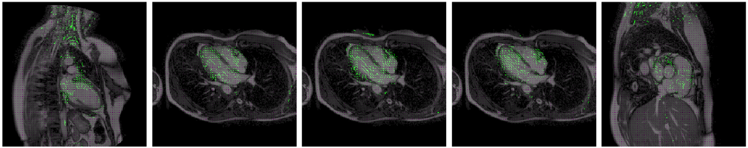
- Every nearest neighbour in the quantised feature space is weighted according to the feature space distance.
- Every image point from the reciprocal image of the quantised feature is weighted according to the distance between pixels.

Optical flow: results

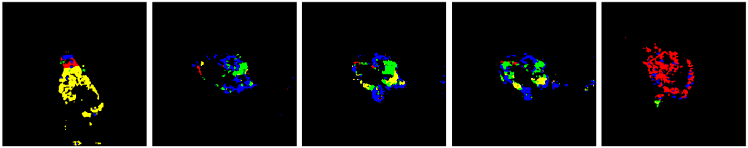
Optical flow: results



Application: Characterization of Cardiac Motion Patterns



(a) Typical results of computing motion estimations for the three planes



(b) shows the flow determined by the different selected clusters on every slice of a cardiac cycle.

[PhD F. Rodríguez, UNAL Bogotá]

Background modelling by the feature space codebook

The codebook of the quantised feature space can be used to model the visual apparence of the background in the methods based on sample and consensus.

The principle is to keep record, for every pixel $\mathbf{x}$, of a collection of M prototypes $\{\mathbf{m}_j(f,\mathbf{x})\}_{j\in\{1,M\}}$, such that every prototype is a word from the quantised feature space (sampling phase).

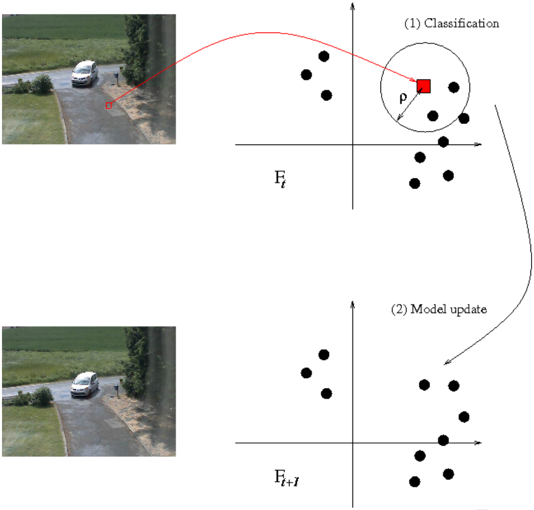
Background subtraction in the feature space

The foreground classification is then performed by counting, for every $\mathbf{x}$ at time t, the number of prototypes that are close enough to $\hat{\mathbf{x}}_{f_t}$ (consensus phase).

$$\begin{aligned} e(f,t,\mathbf{x}) &= 1 \text{ if } |\{j \in \{1,M\}; d^F(\hat{\mathbf{x}}_{f_t}, m_j(f,\mathbf{x})) > \rho\}| > \tau \\ &= 0 \text{ otherwise} \end{aligned}$$

Nota: The same codebook is used for the whole sequence.

The ViBe algorithm in the feature space



Motion detection: results

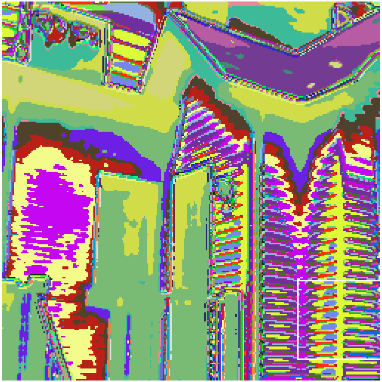
Outline

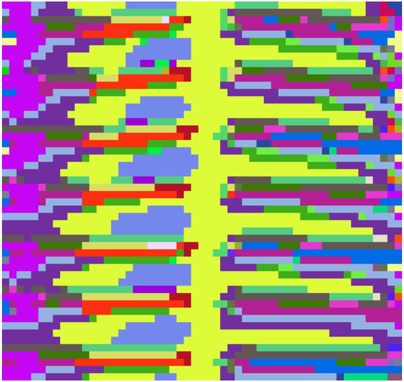
Codebook histograms

The distribution of the words (histogram) of the codebook provides a possible representation for an image or any visual category. See for example [Rubner 99]

Second order statistics of the codebook, i.e. co-occurrence in the image domain of the back-projected words should also be a useful descriptor.

Quantisation of the feature space

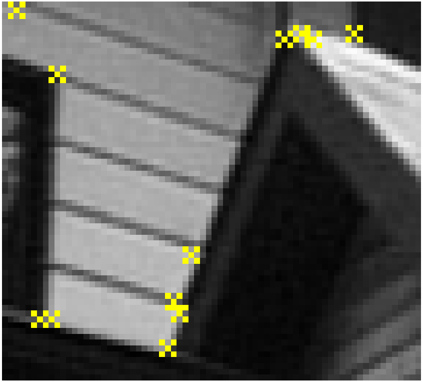




Singularities of the feature space

The isolated points in the feature space (i.e. the feature vectors whose average distance to their K nearest neighbours is the greatest), can be consider as a relevant way to fuse the detection of salient point and the calculation of attached descriptors. See also [Kervrann 08].

Singularities of the feature space



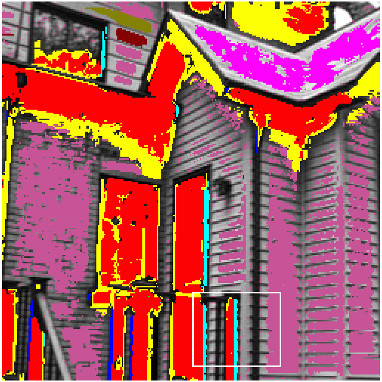
Modes of the feature space

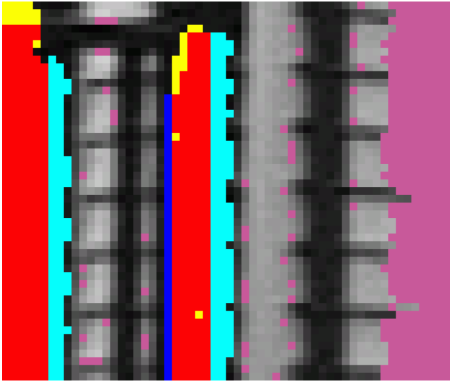
The clusters in the feature space (whose center are the feature vectors whose average distance to their K nearest neighbours is the smallest), are also an interesting way to reduce the visual information to the most relevant features.

The clusters (or modes) are calculated - see [Burman 09] - by (dynamically) defining a topology in the cloud of feature vectors, then labelling the connected component containing the current center point (feature point with smallest distance to its neighbours).

The back-projection of the clusters in the image space correspond to the detection of large homogeneous coulours, long straight edges or regular texture elements.

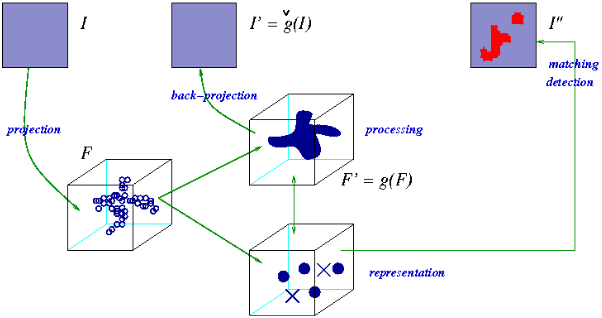
Modes of the feature space



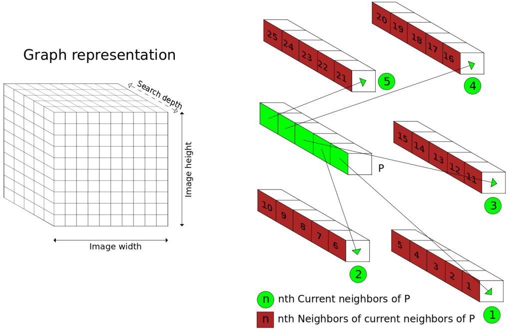


Outline

Overview of the framework



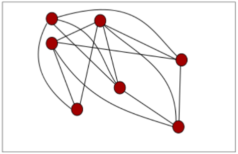
Parallel NN search [Matthieu Garrigues 2010]



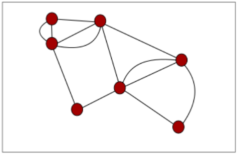
Efficient parallel NN search is developped on GP-GPU: It is based on a volumic representation of the image $\times$ NN spaces, and a recursive search of the NN that allow exponential search in the image space

Parallel NN search [Matthieu Garrigues 2010]

Random initialization of neighbors



For all points p, update p´s neighbors
with neighbors of neighbors of p that
are closer than the current neighbors of p...



... Until the graph remains stable.

Parallel NN search [Matthieu Garrigues 2010]

Bibliography

[Peyré 09] G. PEYRE
Manifold Models for Signals and Images
Computer Vision and Image Understanding 113(2), 249-260. (2009)

[Lindeberg 98] T. LINDEBERG
Feature detection with automatic scale selection
International Journal of Computer Vision 30(2), 77-116. (1998)

[Koenderink 87] J.J. KOENDERINK and A.J. VAN DOORN
Representation of Local Geometry in the Visual System
Biological Cybernetics 55, 367-375. (1987)

Bibliography

- **[Freeman 91]** W.T. FREEMAN and E.H. ADELSON
  The design and use of Steerable Filters
  IEEE Trans. on Pattern Analysis and Machine Intelligence 13(9), 891-906. (1991)
- **[Crosier 10]** M. CROSIER and L.D. GRIFFIN
  Using Basic Image Features for Texture Classification
  International Journal of Computer Vision 88(3), 447-460. (2010)
- **[Rubner 99]** Y. RUBNER and C. TOMASI
  Texture-Based Image Retrieval Without Segmentation
  IEEE International Conference on Computer Vision, Kerkyra, Greece 1018-1024. (1999)

Bibliography

[Mount 97] D.M. MOUNT and S. ARYA
ANN: A Library for Approximate Nearest Neighbor Searching
CGC Workshop on Computational Geometry (1997)
http://www.cs.umd.edu/~mount/ANN/

[Burman 09] P. BURMAN and W. POLONIK
Multivariate mode hunting: Data analytic tools with measures of significance
Journal of Multivariate Analysis 100(6), 1198-1218. (2009)

[Van Vliet 98] L.J. VAN VLIET, I.T. YOUNG and P.W. VERBEEK
Recursive Gaussian derivative filters
Proc. Int. Conf. on Pattern Recognition vol. 1, 509-514. (1998)

Bibliography

[Buades 05] A. BUADES, B. COLL and J.M. MOREL
A non-local algorithm for image denoising
Proc. IEEE Conf. on Computer Vision and Pattern Recognition vol. 2, 60-65. (2005)

[Orchard 08] J. ORCHARD, M. EBRAHIMI and A. WONG
Efficient Non-Local Means Denoising using the SVD
Proc. Int. Conf. on Image Processing 1732-1735. (2008)

[Kervrann 08] C. KERVRANN and J. BOULANGER
Local adaptivity to variable smoothness for exemplar-based image denoising and representation
International Journal of Computer Vision 79(1), 45-69. (2008)

Bibliography

[Barnich 09] O. BARNICH and M. VAN DROGENBROECK
ViBe: a powerful random technique to estimate the background in video sequences
International Conference on Acoustics, Speech, and Signal Processing 945-948. (2009)

[Wang 07] H. WANG and D. SUTER
A consensus-based method for tracking: Modelling background scenario and foreground appearance
Pattern Recognition 40(3), 1091-1105. (2007)

[Kim 04] K. KIM, T.H. CHALIDABHONGSE, D. HARDWOOD and L. DAVIS
Background modeling and subtraction by codebook construction
Proc. Int. Conf. on Image Processing 3061-3064. (2004)

Bibliography

[Manzanera 10a] A. MANZANERA
Image representation and processing through multiscale local jet features
ENSTA-ParisTech Research Report (2010)
http://www.ensta.fr/~manzaner/Publis/LJFeat_AM2010.pdf

[Manzanera 10b] A. MANZANERA
Local Jet Based Similarity for NL-Means Filtering
Proc. Int. Conf. on Pattern Recognition, Istambul, Turkey
2668-2671. (2010)