Patches, a contrario framework and periodicity detection

Valentin De Bortoli[1], Agnès Desolneux[1], Bruno Galerne[2], Arthur Leclaire[1]

[1] *CMLA, ENS Cachan, CNRS, Université Paris-Saclay, 94235 Cachan, France*

[2] *Laboratoire MAP5 (UMR CNRS 8145), Université Paris Descartes, Sorbonne Paris Cité*

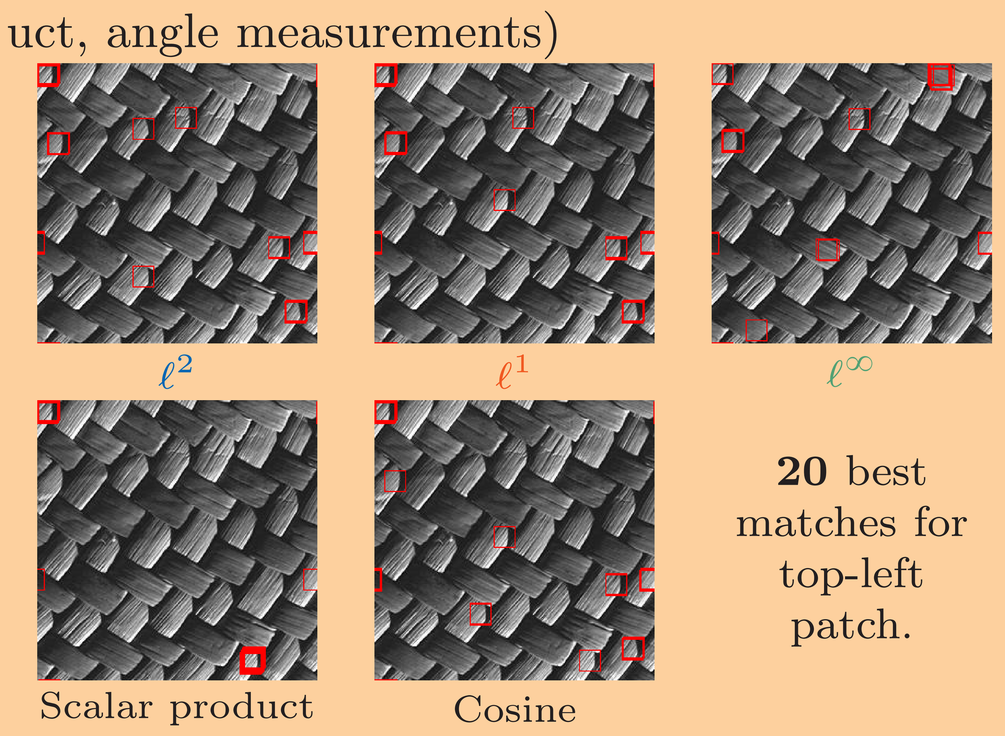
1. Abstract

- Patches are central in image processing [7, 1, 5]
- Introduction of patch comparison strategy with probabilistic guarantees
- Appplication to periodicity detection

2. Comparing patches

Let u be an image defined over $\Omega \subset \mathbb{Z}^2$.
To compare patches in images several comparison functions are available [3]:

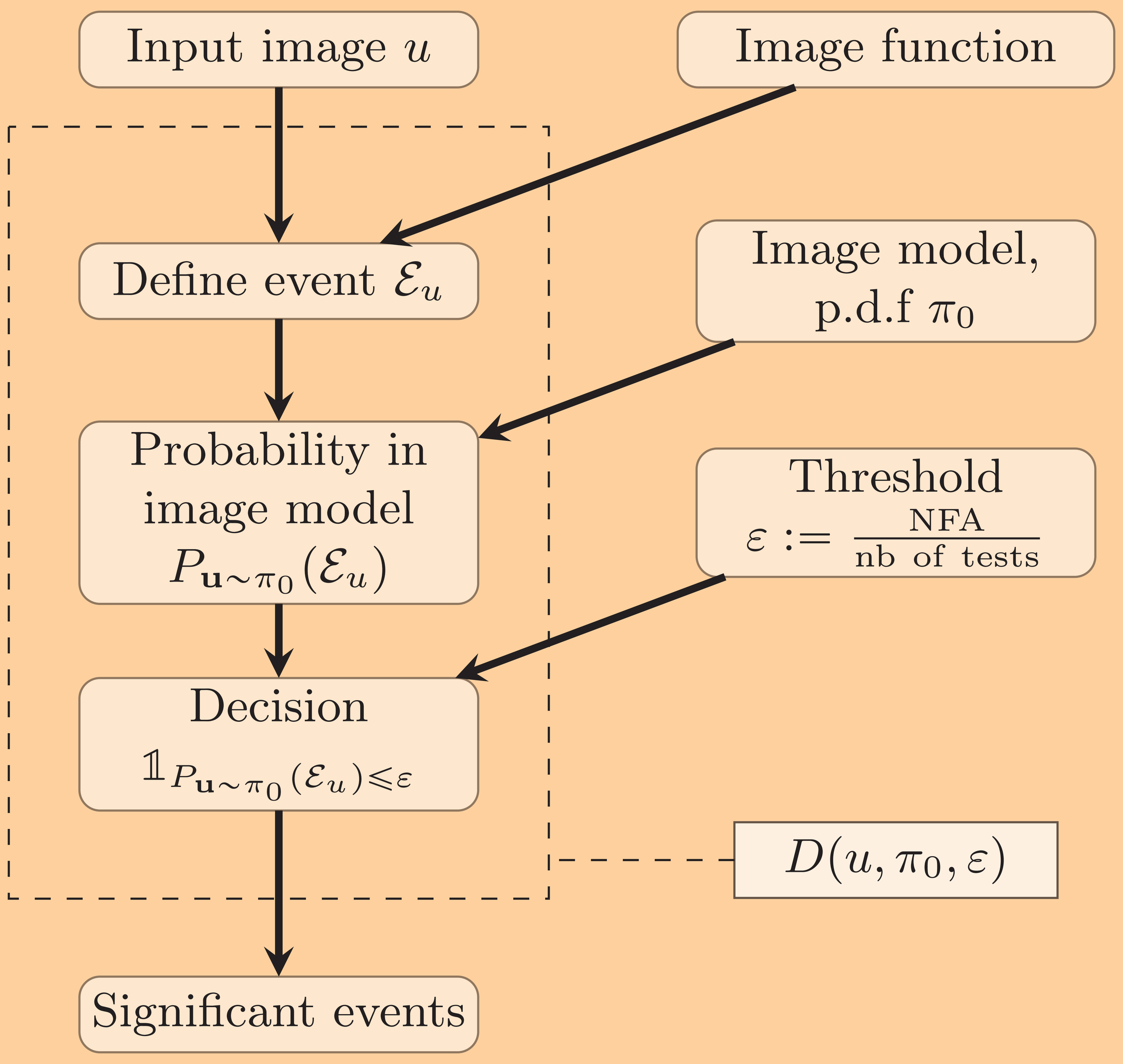
- ℓ^p norms (especially ℓ^1, ℓ^2 and ℓ^∞)
- Directional measurements (Euclidean scalar product, angle measurements)



20 best matches for top-left patch.

3. A contrario framework

Classic *a contrario* framework $D(u, \pi_0, \varepsilon)$, [6]:



Goal: Find most similar patches in image (position maps) using *a contrario* methods.

4. Patch similarity

- s similarity function between patches
- π_0 microtexture model, usually Gaussian [4]
- Two cases of **matching**:
  1. **internal:** $\mathcal{E}_u := s(\mathbf{u}, \mathbf{u}) \leqslant s(u, u)$ (periodicity analysis)
  2. **template:** $\mathcal{E}_u := s(\mathbf{u}, u_0) \leqslant s(u, u_0)$ (texture synthesis)

Proposition *(c.d.f and a contrario)*
We have

$$\mathbb{1}_{P_{\mathbf{u}\sim f}(\mathcal{E}_u)\leqslant\varepsilon} = \mathbb{1}_{F(s(u,u))\leqslant\varepsilon} = \mathbb{1}_{s(u,u)\leqslant F^{-1}(\varepsilon)}$$

♡Take-home 1: **A contrario similarity detection is simply thresholding with adaptive threshold.**

Question: How to compute these c.d.f?

1. Focus on $\ell^2 \to$ Euclidean structure
2. Depending on the similarity function, internal or template matching cases can be computed *exactly*

Theorem *(Internal matching and ℓ^2)*
Let $s(u,u)(x) = \|\mathbb{1}_\omega(u - \tau_x(u))\|_2^2$ where $\omega \subset \Omega$ is a patch index. Suppose $\mathbf{u} \sim \mathcal{N}(0, C)$ then

$$s(\mathbf{u}, \mathbf{u})(x) \sim \sum_{y\in\omega} \lambda_y \xi_y$$

with λ_y eigenvalues of $C_x = 2C - \tau_x C - \tau_{-x} C$ and ξ_y independent χ_2 r.v.

5. Periodicity analysis

Goal: Given u with a periodic pattern P find the underlying lattice $\mathcal{L} = (e_1, e_2)$. Let X be a set of vertices.

Assumption *(Deformed lattice hypothesis)*

$$\forall x \in X, \forall y \in \mathcal{N}_x, \exists (m_{\{x,y\}}, n_{\{x,y\}}) \in \mathbb{Z}^2, x - y = m_{\{x,y\}} e_1 + n_{\{x,y\}} e_2 + \sigma Z_{\{x,y\}}$$

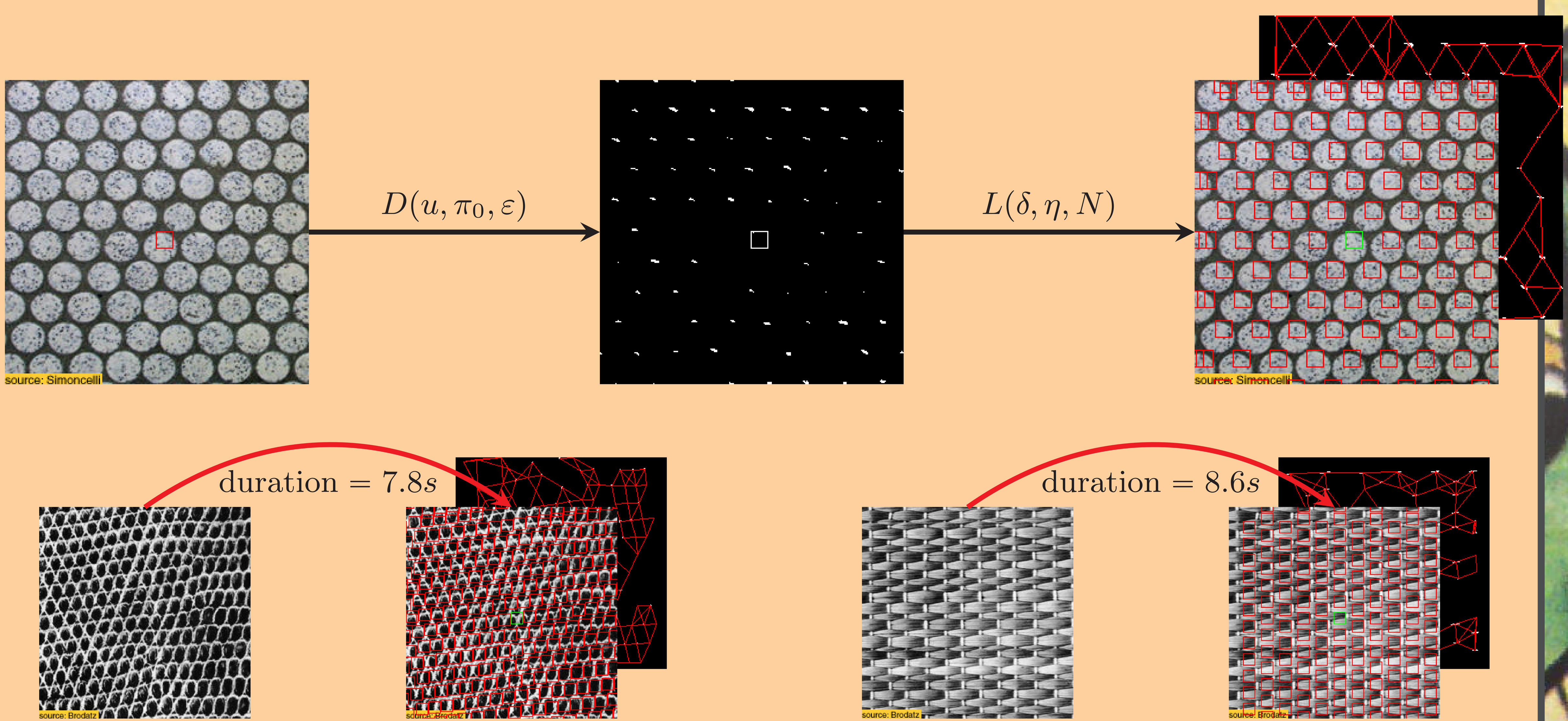
where $\mathcal{N}_x$ is a neighborhood of x and $Z_{\{x,y\}}$ are independent standard Gaussian r.v.

♡Take-home 2: **This hypothesis is naturally translated in a graphical model and reduces to parameters estimation of** $(e_1, e_2, m_{\{x,y\}}, n_{\{x,y\}}, \sigma)$.

$$\underbrace{L(X)}_{\text{log-likelihood}} := -C \log(\sigma^2) - \frac{1}{2\sigma^2} \underbrace{\left(\sum_{x\in X, y\in\mathcal{N}_x} \|(m_{\{x,y\}} e_1 + n_{\{x,y\}} e_2) - (x - y)\|^2 + \overbrace{r(m_{\{x,y\}}, n_{\{x,y\}}, e_1, e_2, \delta, \eta)}^{\text{regularization}} \right)}_{E_{\delta,\eta}(m_{\{x,y\}}, n_{\{x,y\}}, e_1, e_2)}$$

Algorithm 1 Lattice detection – $L(\delta, \eta, N)$

Require: δ, η and N set by user
  for $n = 0$ to $N - 1$ **do**
    $(\tilde{m}_{\{x,y\}}, \tilde{n}_{\{x,y\}}) \leftarrow \operatorname{argmin}_{\mathbb{R}^2} E_{\delta,0}(m_{\{x,y\}}, n_{\{x,y\}}, e_1^n, e_2^n)$
    $(m_{\{x,y\}}^{n+1}, n_{\{x,y\}}^{n+1}) \leftarrow \operatorname{argmin} \left(E_{\delta,0}(\text{round}\left((\tilde{m}_{\{x,y\}}, \tilde{n}_{\{x,y\}})\right), e_1^n, e_2^n), E_{\delta,0}(m_{\{x,y\}}^n, n_{\{x,y\}}^n, e_1^n, e_2^n)\right)$
    $(e_1^{n+1}, e_2^{n+1}) \leftarrow \operatorname{argmin}_{(\mathbb{R}^2)^2} E_{0,\eta}(m_{\{x,y\}}^{n+1}, n_{\{x,y\}}^{n+1}, e_1, e_2)$
  end for



Proposition *(Finite time convergence)*
Under conditions on (δ, η) Algorithm 1 converges in **finite time**. Moreover the log-likelihood is increased at each step.

Problems & remarks:

- Conditions not satisfied in practice, **but** finite time convergence still observed
- Can be written as **EM** algorithm

Conclusion

- Euclidean similarities allow for **fast computation**
- Thresholds for similarity detection are derived from **a contrario** methods
- Periodicity detection algorithm, links with **co-occurence** matrix methods [2]

Bibliography

[1] Antoni Buades, Bartomeu Coll, and J-M. Morel. "A non-local algorithm for image denoising". In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 2. IEEE. 2005, pp. 60–65.

[2] Richard W. Conners and Charles A. Harlow. "Toward a structural textural analyzer based on statistical methods". In: *Computer Graphics and Image Processing* 12.3 (1980), pp. 224–256.

[3] Charles-Alban Deledalle, Loïc Denis, and Florence Tupin. "How to compare noisy patches? Patch similarity beyond Gaussian noise". In: *International journal of computer vision* 99.1 (2012), pp. 86–102.

[4] Bruno Galerne, Yann Gousseau, and Jean-Michel Morel. "Random phase textures: Theory and synthesis". In: *IEEE Transactions on image processing* 20.1 (2011), pp. 257–267.

[5] Kaiming He and Jian Sun. "Image completion approaches using the statistics of similar patches". In: *IEEE transactions on pattern analysis and machine intelligence* 36.12 (2014), pp. 2423–2435.

[6] David Mumford and Agnès Desolneux. *Pattern theory: the stochastic analysis of real-world signals*. CRC Press, 2010.

[7] Lara Raad, Agnès Desolneux, and Jean-Michel Morel. "Conditional Gaussian models for texture synthesis". In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2015, pp. 474–485.