Modèles statistiques pour l'image
Patch-based Image Processing

Julie Digne

LIRIS - CNRS

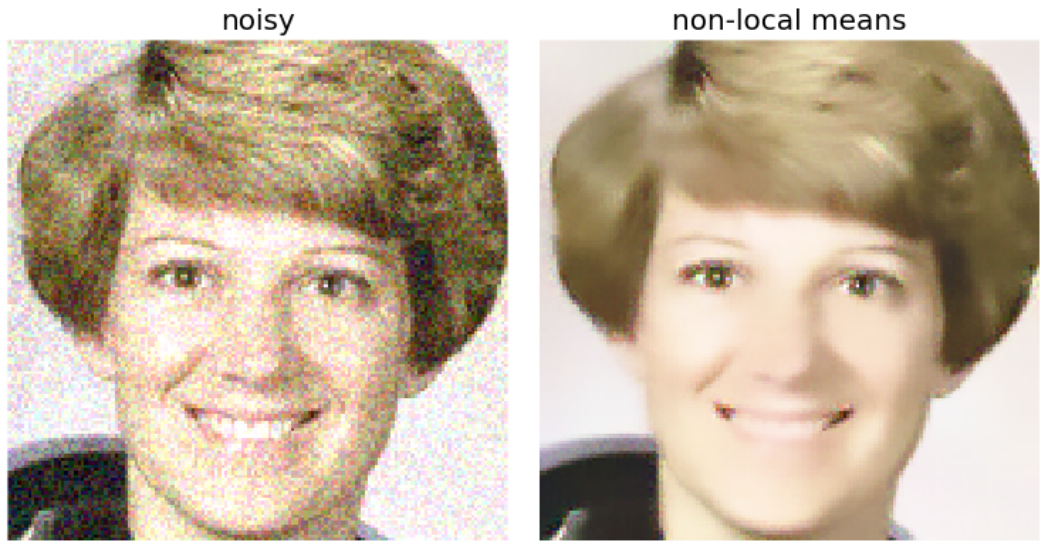
18/09/2024

Outline

Patch-based processing

- Consider patches instead of pixels

Similarity Analysis: Non Local Means [Buadès et al. 2005]



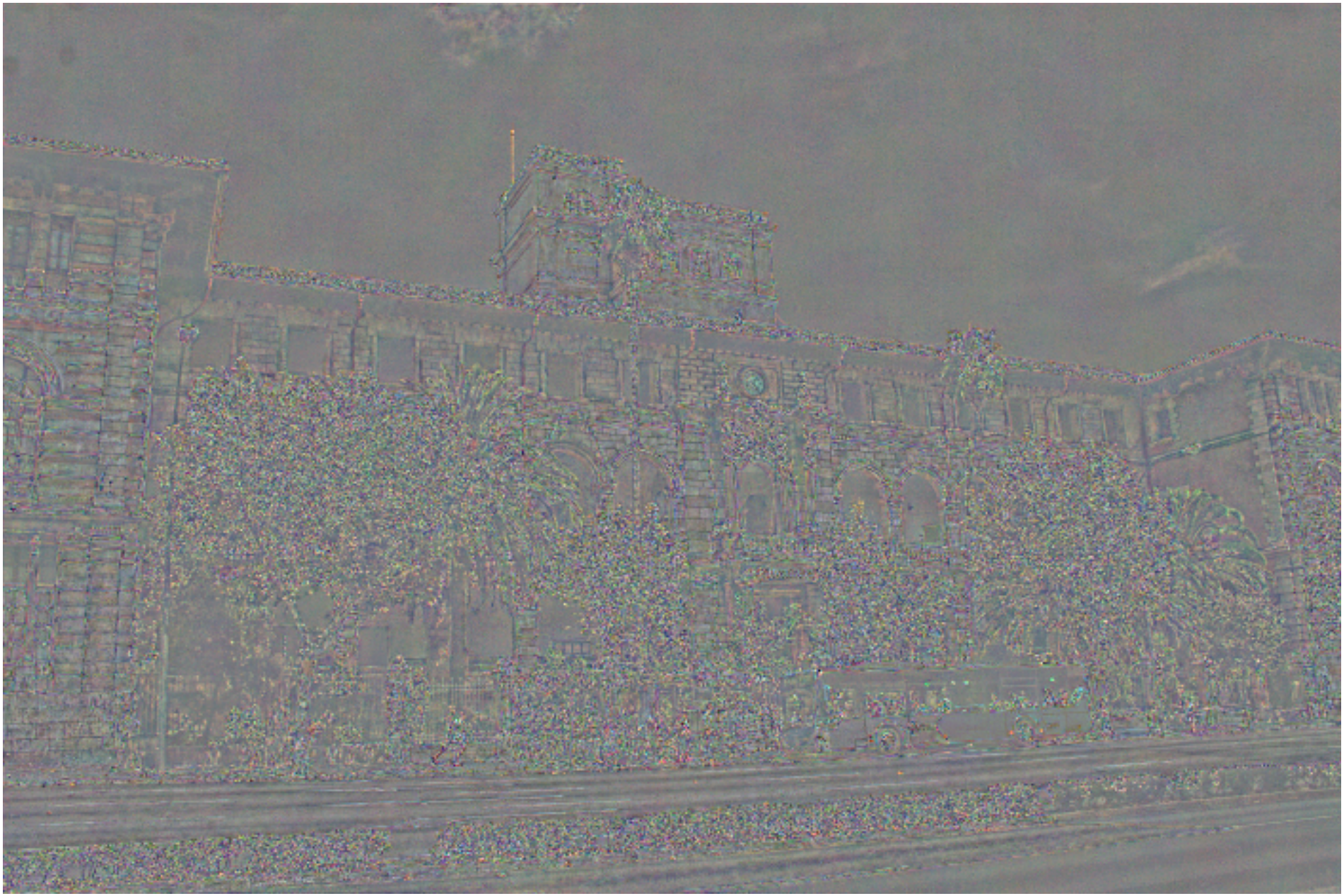
- Idea: denoise a point by comparing it to similar neighborhoods
- Compute local patch $P(p)$ around each point p
- Similarity measure between two points: $w(p,q) = \exp -\frac{dist(P(p),P(q))^2}{\sigma}$
- Update of the image :

$$I_{new}(p) = \frac{\sum_{q \in I} w(p,q) I(q)}{\sum_{q \in I} w(p,q)}$$

Example

Example

Example



Initial image

Noisy image

Gaussian filter result

Gaussian filter result

Gaussian filter result

Median result

Median result

NLmeans result

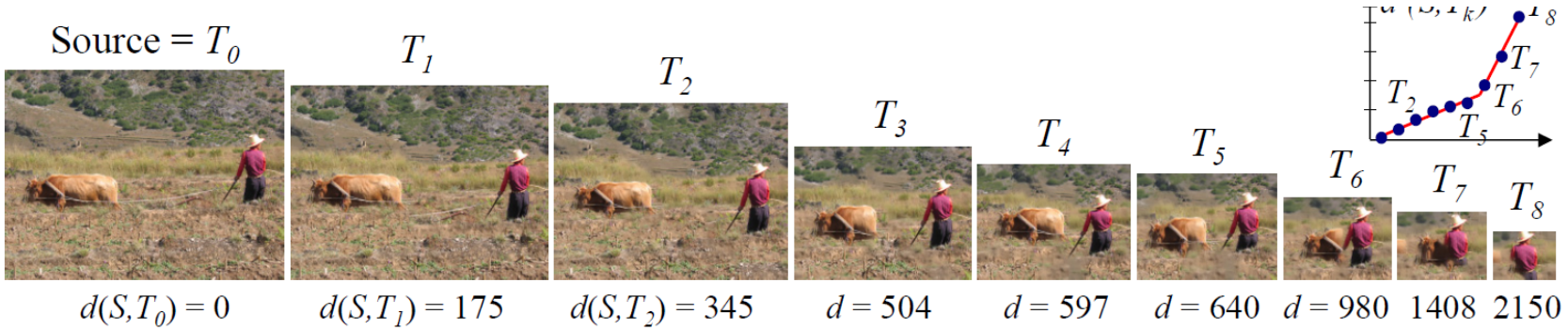
Comparison

Outline

Visual Summary

Goal

Produce a smaller image that summarizes the content of the larger image



[Simakov 2008]

Comparing two images

Bidirectional Distance (BDS) [Simakov 2008]

Source image S, target image T:

$$d_{BDS}(S,T) = \frac{1}{N_S}\sum_{s\subset S}\min_{t\subset T} D(s,t) + \frac{1}{N_T}\sum_{t\subset T}\min_{s\subset S} D(t,s)$$

where s and t are patches of fixed size of S and T. D is the sum of squared difference between patches.

Comparing two images

Bidirectional Distance (BDS) [Simakov 2008]

Source image S, target image T:

$$d_{BDS}(S,T) = \frac{1}{N_S}\sum_{s\subset S}\min_{t\subset T} D(s,t) + \frac{1}{N_T}\sum_{t\subset T}\min_{s\subset S} D(t,s)$$

where s and t are patches of fixed size of S and T. D is the sum of squared difference between patches.

- Completeness term

Comparing two images

Bidirectional Distance (BDS) [Simakov 2008]

Source image S, target image T:

$$d_{BDS}(S,T) = \frac{1}{N_S}\sum_{s\subset S}\min_{t\subset T} D(s,t) + \frac{1}{N_T}\sum_{t\subset T}\min_{s\subset S} D(t,s)$$

where s and t are patches of fixed size of S and T. D is the sum of squared difference between patches.

- Completeness term
- Coherence term

Comparing two images

Bidirectional Distance (BDS) [Simakov 2008]

Source image S, target image T:

$$d_{BDS}(S,T) = \frac{1}{N_S}\sum_{s\subset S}\min_{t\subset T} D(s,t) + \frac{1}{N_T}\sum_{t\subset T}\min_{s\subset S} D(t,s)$$

where s and t are patches of fixed size of S and T. D is the sum of squared difference between patches.

- Completeness term
- Coherence term

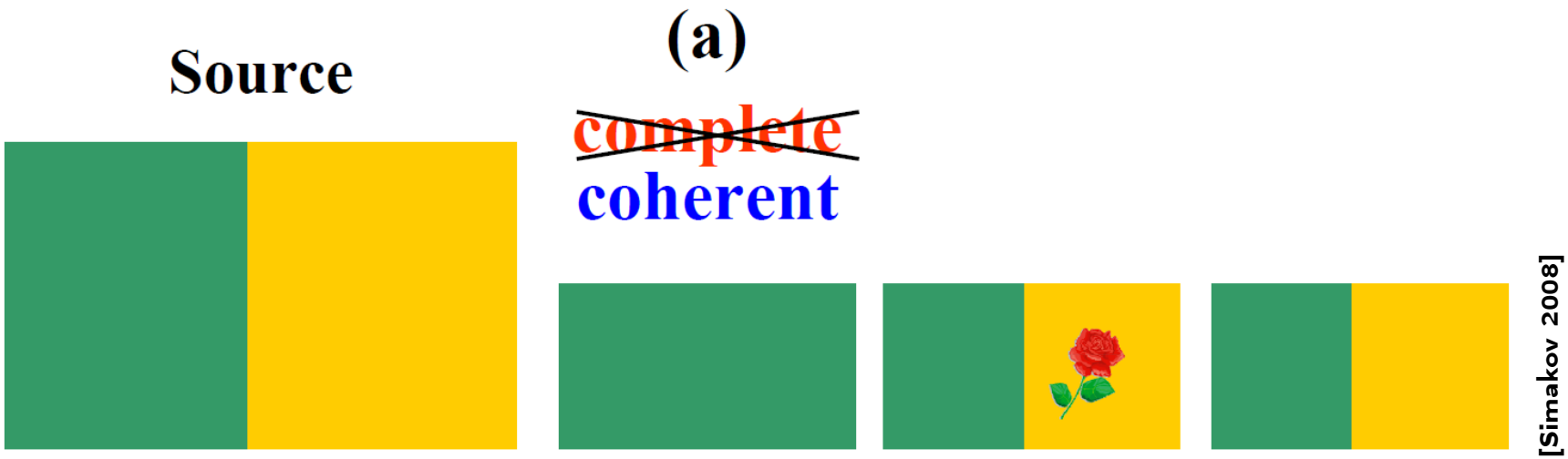
Reconstruction

Starting from an initial guess T_0 for T, build an image iteratively as the minimizer T of $d_{BDS}(S,T)$

Similarity distance

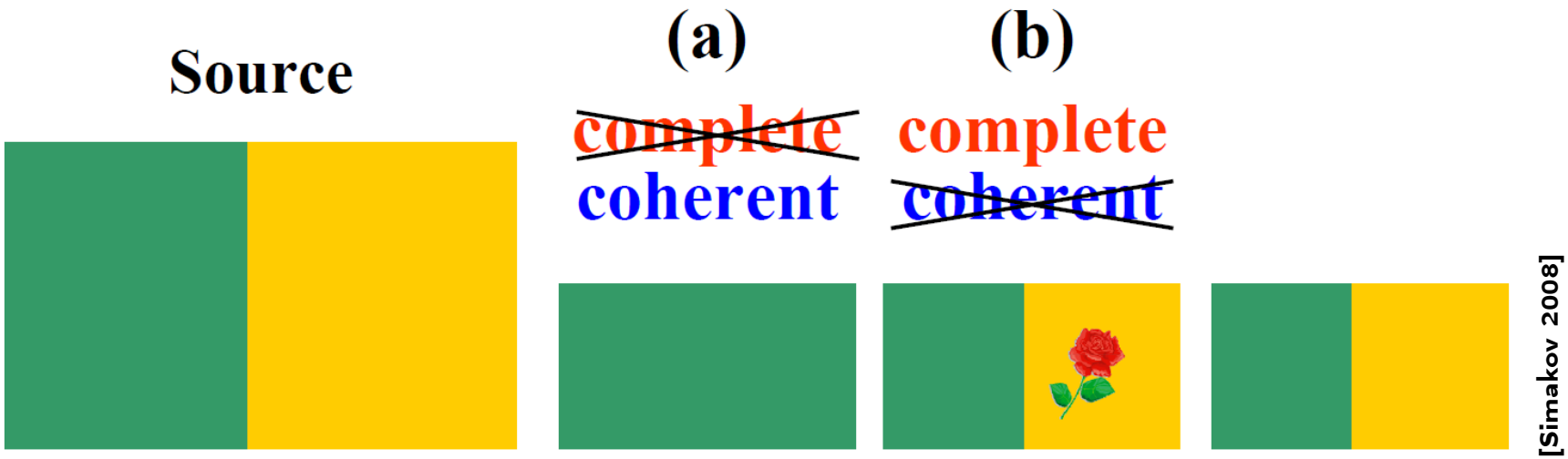
[Simakov 2008]

Similarity distance



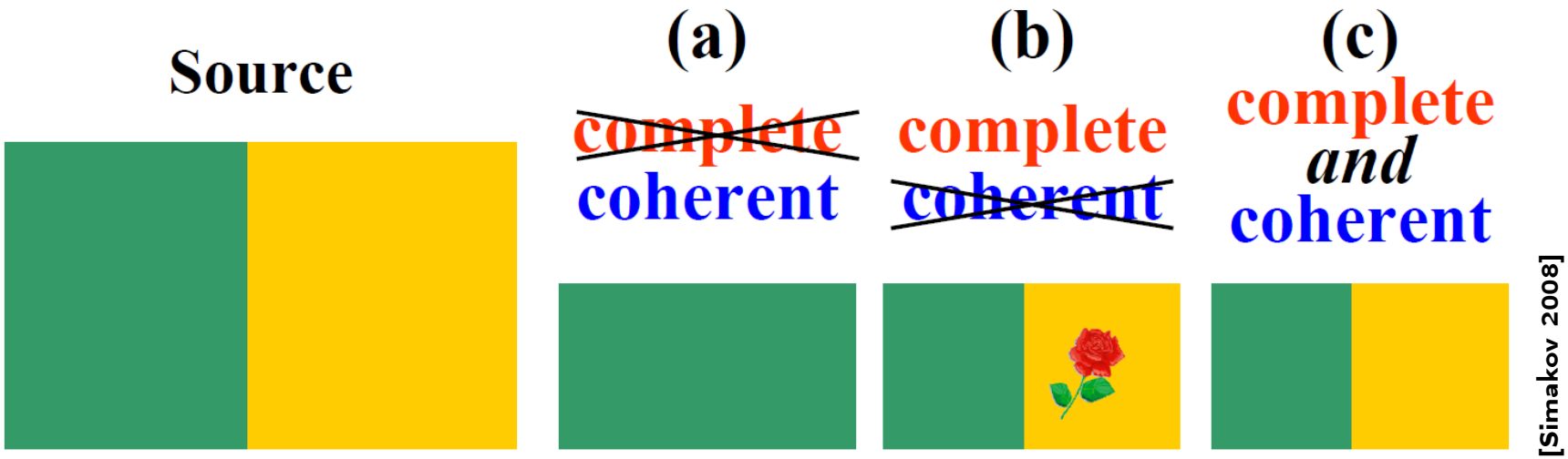
[Simakov 2008]

Similarity distance



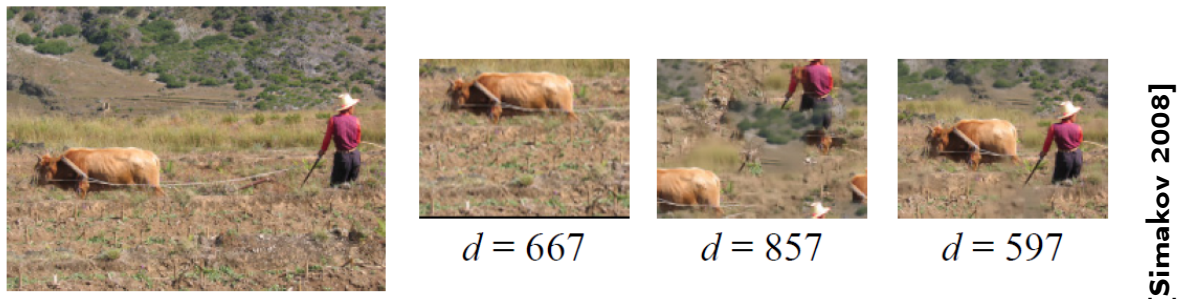
[Simakov 2008]

Similarity distance



[Simakov 2008]

Similarity distance



[Simakov 2008]

A three steps algorithm

1. For each patch of S find its closest patch on T

A three steps algorithm

1. For each patch of S find its closest patch on T
2. For each patch of T find its closest patch on S

A three steps algorithm

1. For each patch of S find its closest patch on T
2. For each patch of T find its closest patch on S
3. Aggregate color of nearest patches and update colors of T

A three steps algorithm

1. For each patch of S find its closest patch on T
2. For each patch of T find its closest patch on S
3. Aggregate color of nearest patches and update colors of T

Steps 1 - 2

The two first steps consist in applying nearest patch search. It needs to be done *efficiently*.

Aggregation Step: contribution of a pixel to the coherence measure

- Let q be a pixel of T, q lies inside m neighboring patches $Q_1, Q_2, \cdots, Q_m$

Aggregation Step: contribution of a pixel to the coherence measure

- Let q be a pixel of T, q lies inside m neighboring patches $Q_1, Q_2, \cdots, Q_m$
- These patches are matched to $P_1, P_2, \cdots, P_m$

Aggregation Step: contribution of a pixel to the coherence measure

- Let q be a pixel of T, q lies inside m neighboring patches $Q_1, Q_2, \cdots, Q_m$
- These patches are matched to $P_1, P_2, \cdots, P_m$
- The positions corresponding to q in $P_1, P_2, \cdots, P_m$ are $p_1, \cdots, p_m$

Aggregation Step: contribution of a pixel to the coherence measure

- Let q be a pixel of T, q lies inside m neighboring patches $Q_1, Q_2, \cdots, Q_m$
- These patches are matched to $P_1, P_2, \cdots, P_m$
- The positions corresponding to q in $P_1, P_2, \cdots, P_m$ are $p_1, \cdots, p_m$

Contribution

$$\frac{1}{N_T} \sum_{i=1}^{m} \|S(p_i) - T(q)\|_2^2$$

Aggregation Step: contribution of a pixel to the completeness measure

- Let q be a pixel of T,

Aggregation Step: contribution of a pixel to the completeness measure

- Let q be a pixel of T,
- q lies inside n neighboring patches $\hat{Q}_1, \hat{Q}_2, \cdots, \hat{Q}_n$ that are the nearest patch to some patches of S $\hat{P}_1, \hat{P}_2, \cdots \hat{P}_n$

Aggregation Step: contribution of a pixel to the completeness measure

- Let q be a pixel of T,
- q lies inside n neighboring patches $\hat{Q}_1, \hat{Q}_2, \cdots, \hat{Q}_n$ that are the nearest patch to some patches of S $\hat{P}_1, \hat{P}_2, \cdots \hat{P}_n$
- The positions corresponding to q in $\hat{P}_1, \hat{P}_2, \cdots, \hat{P}_m$ are $\hat{p}_1, \cdots, \hat{p}_m$

Aggregation Step: contribution of a pixel to the completeness measure

- Let q be a pixel of T,
- q lies inside n neighboring patches $\hat{Q}_1, \hat{Q}_2, \cdots, \hat{Q}_n$ that are the nearest patch to some patches of S $\hat{P}_1, \hat{P}_2, \cdots \hat{P}_n$
- The positions corresponding to q in $\hat{P}_1, \hat{P}_2, \cdots, \hat{P}_m$ are $\hat{p}_1, \cdots, \hat{p}_m$

Contribution

$$\frac{1}{N_S} \sum_{i=1}^{n} \|S(\hat{p}_i) - T(q)\|_2^2$$

Color update

Color Update

The best $T(q)$ should minimize:

$$\frac{1}{N_S}\sum_{i=1}^{n}\|S(\hat{p}_i) - T(q)\|_2^2 + \frac{1}{N_T}\sum_{i=1}^{m}\|S(p_i) - T(q)\|_2^2$$

Color update

Color Update

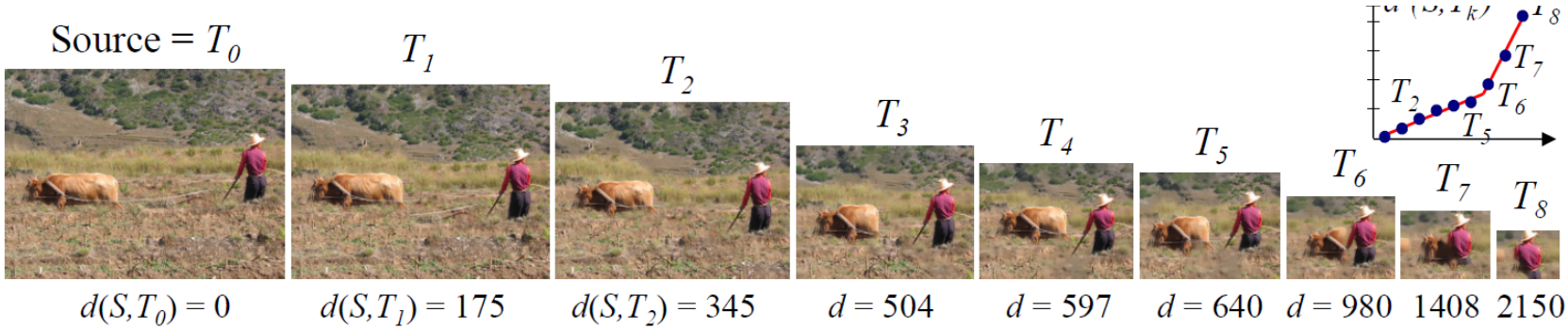
The best $T(q)$ should minimize:

$$\frac{1}{N_S}\sum_{i=1}^{n}\|S(\hat{p}_i) - T(q)\|_2^2 + \frac{1}{N_T}\sum_{i=1}^{m}\|S(p_i) - T(q)\|_2^2$$

Color Update

$$T(q) = \frac{\frac{1}{N_S}\sum_{i=1}^{n} S(\hat{p}_i) + \frac{1}{N_T}\sum_{i=1}^{m} S(p_i)}{\frac{m}{N_T} + \frac{n}{N_S}}$$

Visual Summary



[Simakov 2008]

Gradual resizing

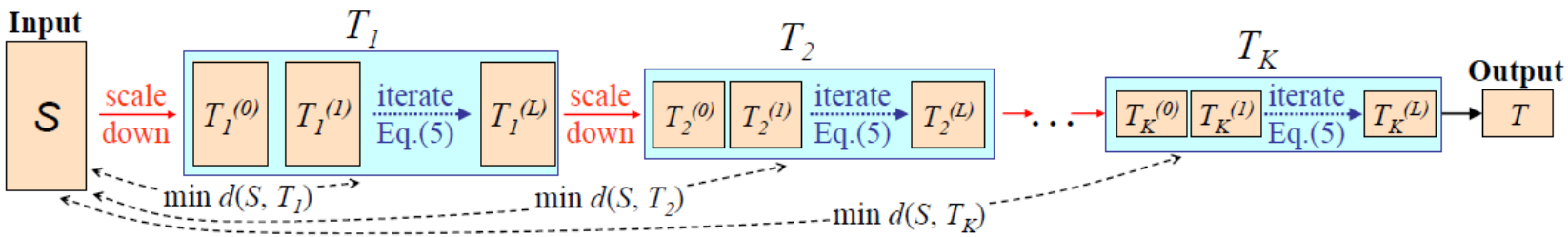
- When the target has a very different size from the source: what is a good initial guess?

Gradual resizing

- When the target has a very different size from the source: what is a good initial guess?
- Iterative process: downsample the image and apply the reconstruction

Gradual resizing

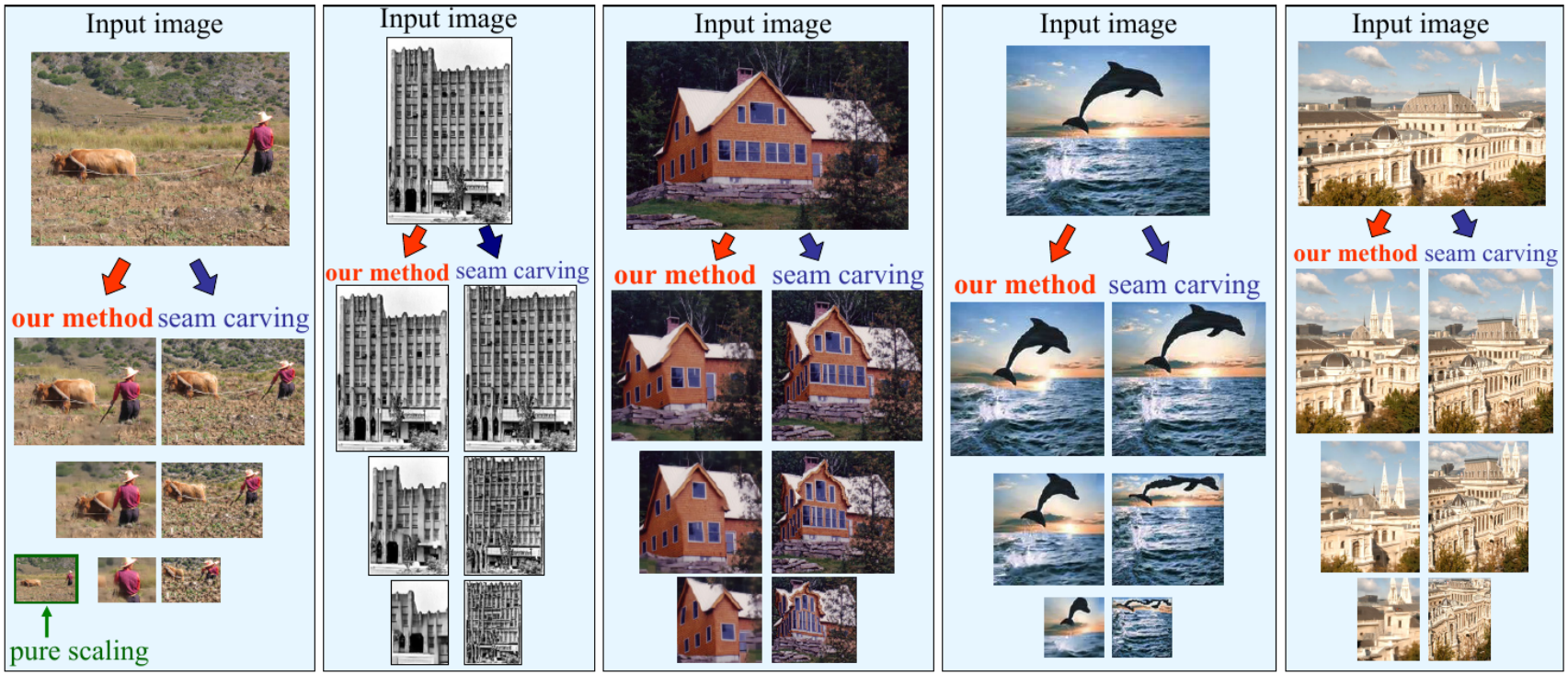
- When the target has a very different size from the source: what is a good initial guess?
- Iterative process: downsample the image and apply the reconstruction



[Simakov 2008]

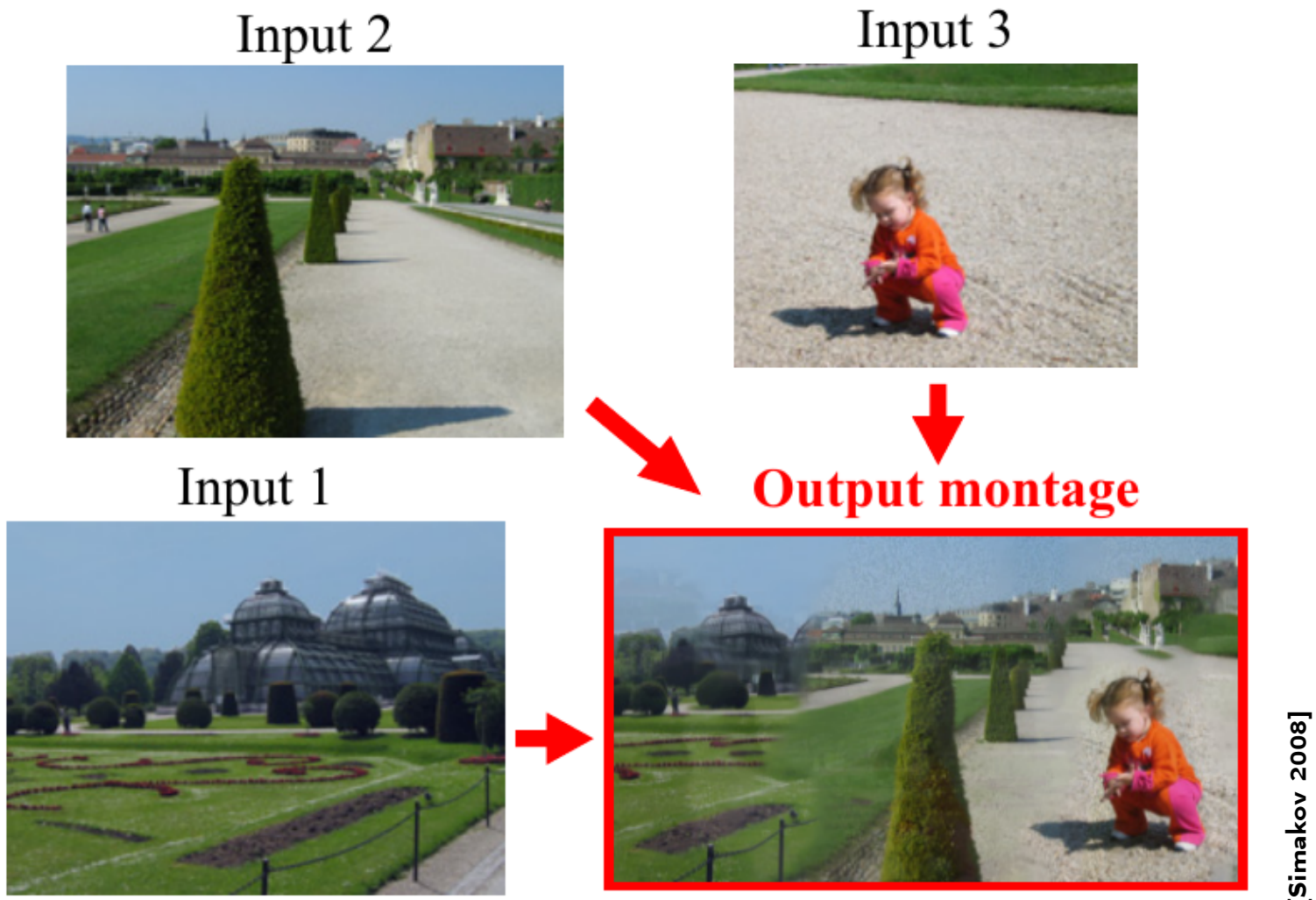
video

Visual Summary



[Simakov 2008]

Montage



[Simakov 2008]

Synthesis

Input

Bigger output (Synthesis)

[Simakov 2008]

Key ingredient for all these methods

Requirement

A fast method to find similar patches

- Naive way: traverse the whole image at each query

Key ingredient for all these methods

Requirement

A fast method to find similar patches

- Naive way: traverse the whole image at each query
- Better: put all patches in a search structure

Key ingredient for all these methods

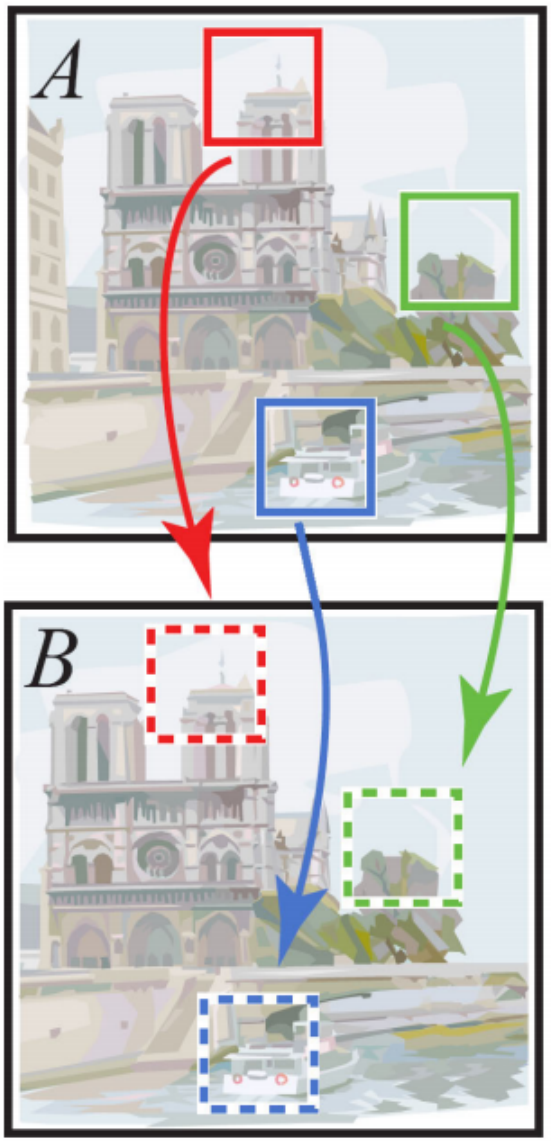
Requirement

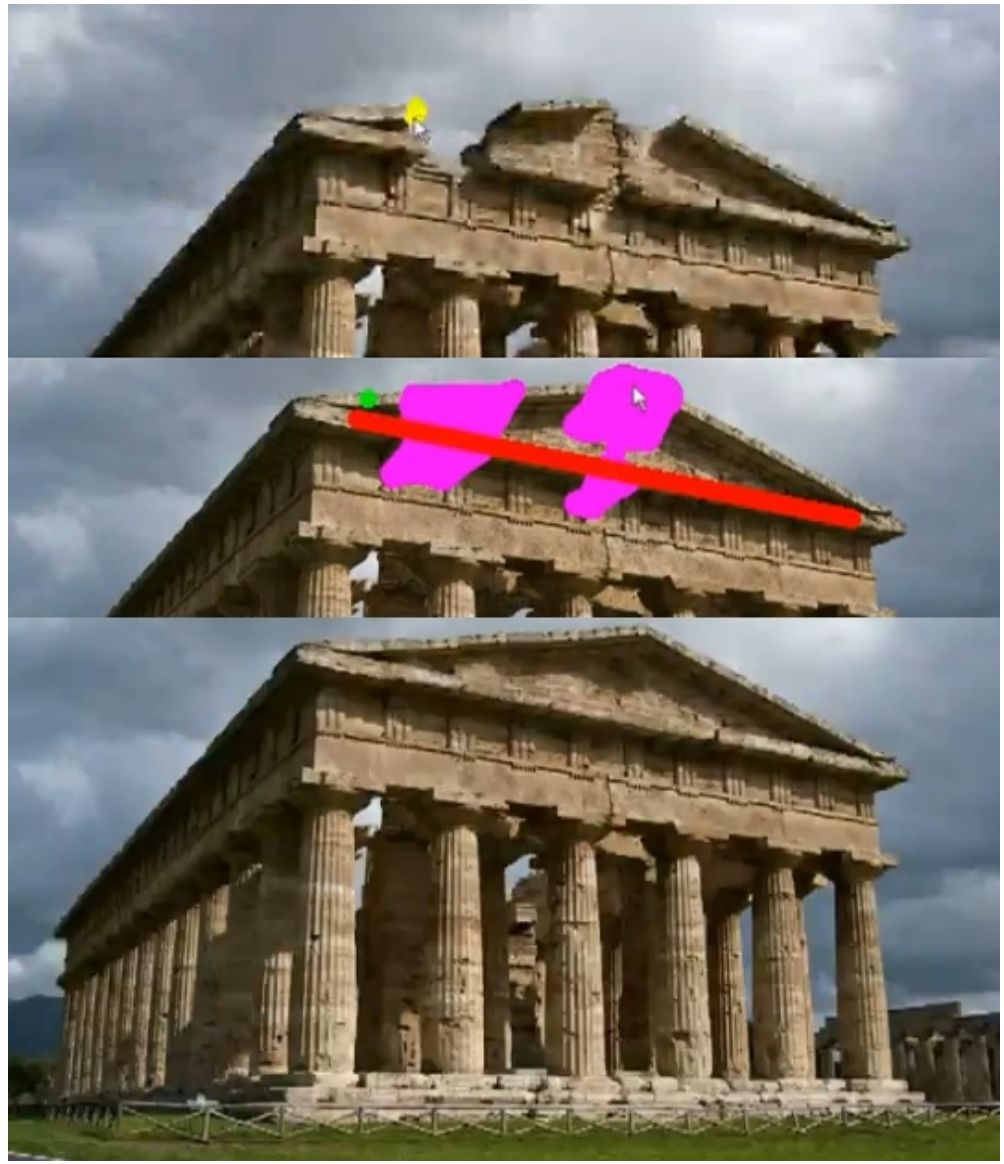
A fast method to find similar patches

- Naive way: traverse the whole image at each query
- Better: put all patches in a search structure
- Even better: the patch match algorithm

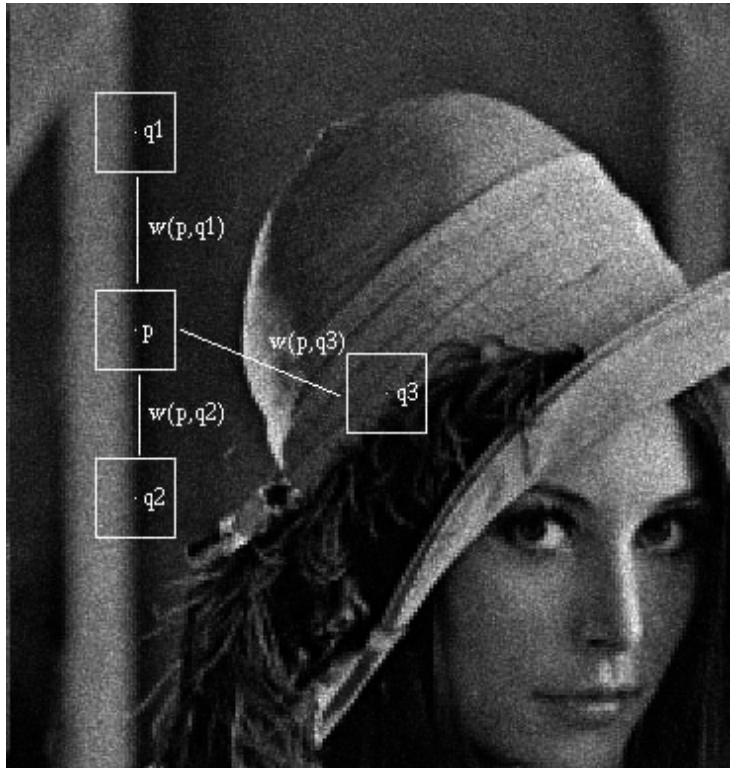
Outline

Patch Match [Barnes 2009]





Similar patches



[Buadès 2005]

Similarity distance

The similarity distance between two patches p_A, p_B of size $n \times n$ is computed as $\sum_{1 \leq i,j \leq n} \|p_A(i,j) - p_B(i,j)\|_2^2$.

Similarity

Two patches are considered as similar is their similarity distance is small.

Patch Match

Goal

Given an image A and an image B find *efficiently* for all patches of image A an approximate nearest patch of image B.

Patch Match

Goal

Given an image A and an image B find *efficiently* for all patches of image A an approximate nearest patch of image B.

Patch Match Principle

Assume we have found a patch p_B of B corresponding to a given patch p_A of A, assume we have a patch p'_A located close to p_A in image A, then its corresponding patch p'_B has a high probability to lie close to p_B

- Look for p'_B close to p_B.

What if?

- If we have an initial corresponding pairs (p_A, p_B) then the search is made easier

What if?

- If we have an initial corresponding pairs (p_A, p_B) then the search is made easier
- However: How can we find an initial pair?

What if?

- If we have an initial corresponding pairs (p_A, p_B) then the search is made easier
- However: How can we find an initial pair?
- However: Should we start with a single initial pair? Why not many?

What if?

- If we have an initial corresponding pairs (p_A, p_B) then the search is made easier
- However: How can we find an initial pair?
- However: Should we start with a single initial pair? Why not many?

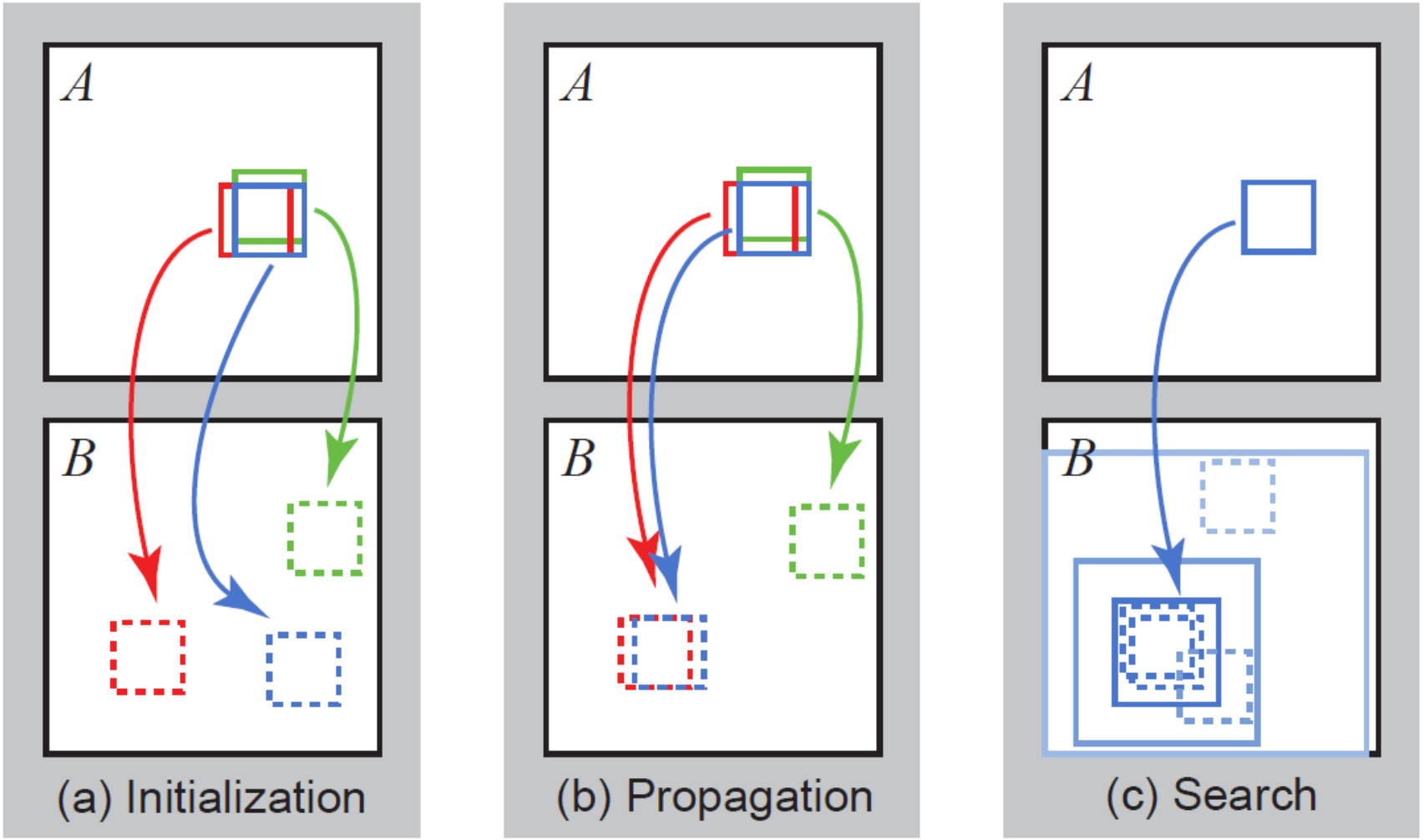
Notation

Let p_A be a patch centered at a in image A and p_B a patch centered at b in image B. We define an offset vector $f(a)$ as $f(a) = b - a$. The set of all offset vectors is called the Nearest Neighbor Field (NNF).

Algorithm

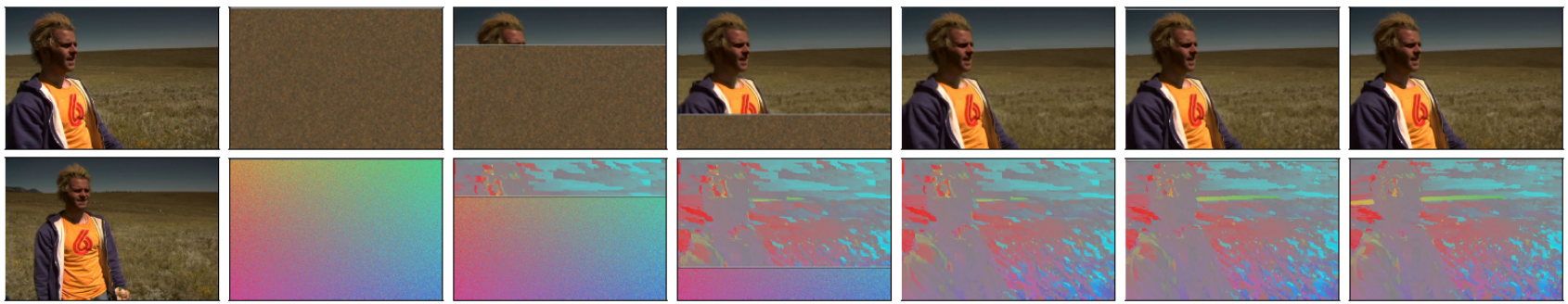
1. Initialize the NNF with random vectors
2. **Propagation:** for $i = 1 \cdots M$, for $j = 1 \cdots M$
   1. Evaluate the offset $f(i-1, j)$, $f(i-1, j-1)$, $f(i-1, j+1)$ and $f(i, j-1)$
   2. If one of them is better than $f(i, j)$ replace $f(i, j)$ with it.
3. **Randomization:** For all (i, j), draw a random offset w, if w is better than $f(i, j)$ set $f(i, j) = w$

Algorithm



[Barnes 2009]

Algorithm



(a) originals (b) random (c) $\frac{1}{4}$ iteration (d) $\frac{3}{4}$ iteration (e) 1 iteration (f) 2 iterations (g) 5 iterations

[Barnes 2009]

Probabilistic analysis

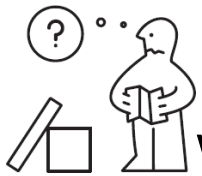
Exercise

Assume we have two images of size M and assign randomly patches of image A to patches of image B (+ unicity of the correspondence)

Probabilistic analysis

Exercise

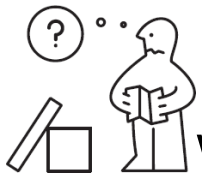
Assume we have two images of size M and assign randomly patches of image A to patches of image B (+ unicity of the correspondence)

- 
  What is the probability that at least one patch is indeed paired to its corresponding patch?

Probabilistic analysis

Exercise

Assume we have two images of size M and assign randomly patches of image A to patches of image B (+ unicity of the correspondence)

- 
What is the probability that at least one patch is indeed paired to its corresponding patch?

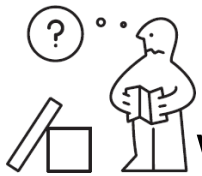
Simplification

Assume that a pair is correct if a patch is assigned to a patch that is spatially close (in a neighborhood of size C) to its true correspondence

Probabilistic analysis

Exercise

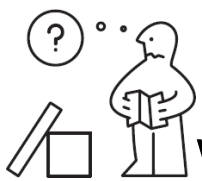
Assume we have two images of size M and assign randomly patches of image A to patches of image B (+ unicity of the correspondence)



- What is the probability that at least one patch is indeed paired to its corresponding patch?

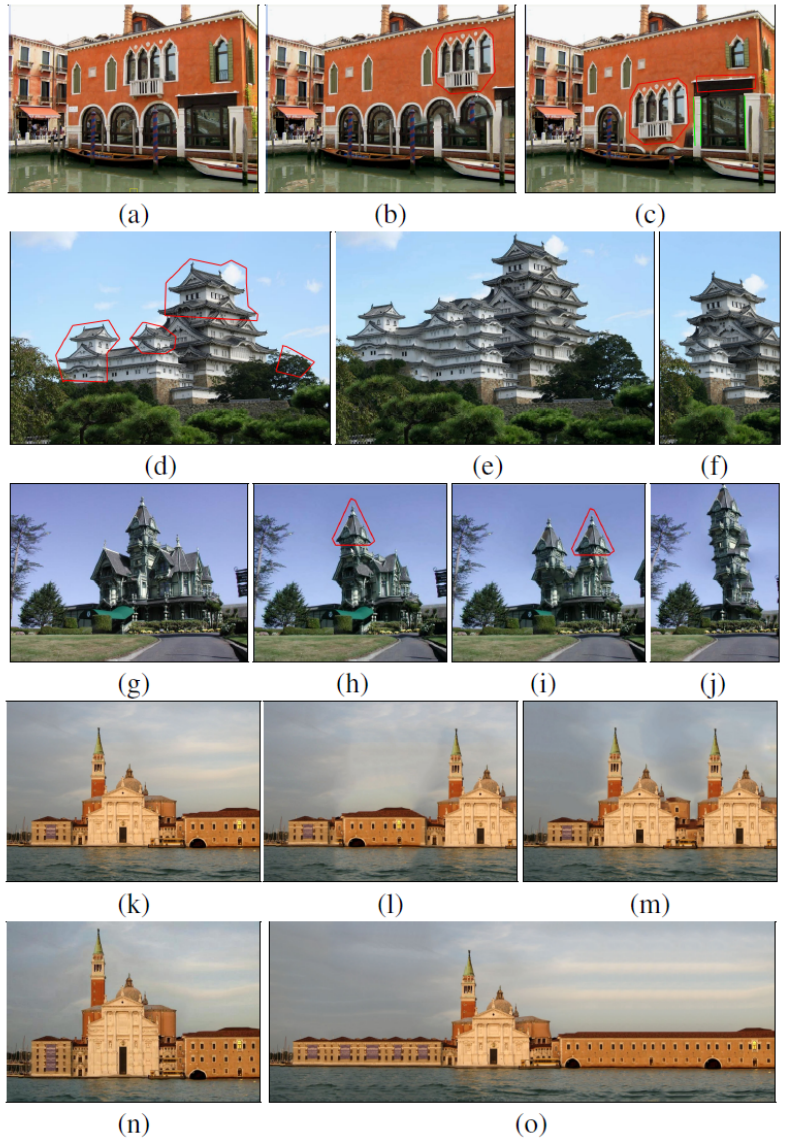
Simplification

Assume that a pair is correct if a patch is assigned to a patch that is spatially close (in a neighborhood of size C) to its true correspondence



- What is the probability that at least one patch is paired to an approximate corresponding patch?

Reshuffling Application

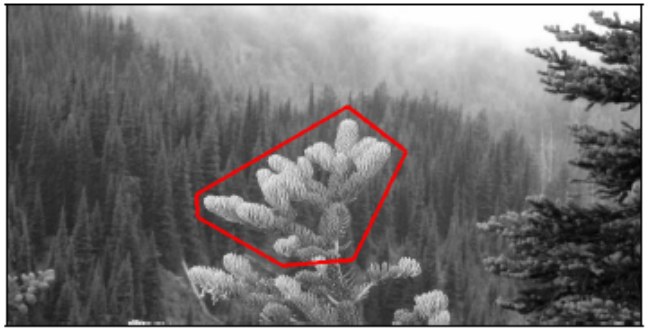


[Barnes 2009]

Deformation Application

(a) building marked by user

(b) scaled up, preserving texture



(c) bush marked by user

(d) scaled up, preserving texture.

[Barnes 2009]