PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing

Connelly Barnes¹, Eli Shechtman²,³, Adam Finkelstein¹, Dan B Goldman²

¹Princeton University, ²Adobe Systems, ³University of Washington
PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing

Connelly Barnes\(^1\), Eli Shechtman\(^{2,3}\), Adam Finkelstein\(^1\), Dan B Goldman\(^2\)

\(^1\)Princeton University, \(^2\)Adobe Systems, \(^3\)University of Washington
PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing

(a) Original  (b) Inpainting  (c) Retarget  (d) Reshuffle

Connelly Barnes¹, Eli Shechtman²,³, Adam Finkelstein¹, Dan B Goldman²

¹Princeton University, ²Adobe Systems, ³University of Washington
Roadmap

Motivation

PatchMatch Algorithm

Interactive Editing
Traditional Photo Editing

(played back at 10x real-time)
Higher Level Editing

Retargeting [Avidan ‘07]

Hole filling [Sun ‘05]

Reshuffling [Simakov ‘08], [Cho ‘08]
User Interaction

Retargeting [Avidan ‘07]

Hole filling [Sun ‘05]

Reshuffling [Simakov ‘08], [Cho ‘08]
Non-Parametric Patch Sampling

Image retargeting [Simakov et al. ‘08]
(5 minutes per 250x200 input image)
Non-Parametric Patch Sampling

Image reshuffling/collage [Simakov et al. ‘08]
(5 minutes per 250x200 input image)
Iterative Optimization Methods

Retargeting and Reshuffling
[Simakov '08]

Texture Synthesis
[Kwatra '05]

Hole Filling
[Wexler '04]
Hole Filling

Repeat

- Initialize hole
- Find NN inside → out
- Update image

[Wexler ‘04]
Hole Filling

Initialize hole → Find NN inside → out → Update image

[Wexler '04]
Retargeting / Reshuffling

Repeat

Initial guess \(\rightarrow\) Find NN \(A \leftrightarrow B\) \(\rightarrow\) Update output

[Simakov ‘08]
Retargeting / Reshuffling

Initial guess → Find NN (A ↔ B) → Update output

Slow

[Simakov ‘08]
Related Work

- **kd-tree with PCA**
  
  [Hertzmann '01]

- **Propagation**
  
  [Ashikhmin '01]

- **k-coherence**
  
  [Tong '02]
Problem
Intuition for PatchMatch
Intuition for PatchMatch
Intuition for PatchMatch
Intuition for PatchMatch
Intuition for PatchMatch
Nearest Neighbor Field

\[ f(a) = b - a \]
Random Initialization

\( A \quad \overset{\text{arrows}}{\longrightarrow} \quad B \)
After propagation:

\[ f(x, y) = \arg\min_{D} \{ \text{current, left, above} \} \]
After propagation:

\[ f(x, y) = \arg\min_D \{ \text{current, left, above} \} \]
Propagation
Random Search

$A$

$B$
Random Search

Box width: $w$
(image width)
Random Search

$A$

$B$

Box width: $aw$
Random Search

Box width: $a^2w$
Random Search

Box width: 1 pixel
After propagation and search:

\[ f(x, y) = \arg\min_D \{ \text{candidate offsets} \} \]
Propagation Only

First Pass

Image A

Correspondence Vectors
(red: x, blue: y)

Image B

Reconstruction of image A using patches from image B
Propagation Only

Correspondence Vectors (red: x, blue: y)

Image A

Reconstruction of image A using patches from image B

Second Pass
Propagation Only

Correspondence Vectors (red: x, blue: y)

Reconstruction of image A using patches from image B
Random Search Only

First Pass

Correspondence Vectors (red: x, blue: y)

Reconstruction of image A using patches from image B
Random Search Only

Second Pass

Image A

Correspondence Vectors
(red: x, blue: y)

Image B

Reconstruction of image A
using patches from image B
Random Search Only

Correspondence Vectors (red: x, blue: y)

Reconstruction of image A using patches from image B

Third Pass
Convergence

First Pass

Image A

Correspondence Vectors (red: x, blue: y)

Reconstruction of image A using patches from image B

Image B
Convergence

Propagation
Random Search
PatchMatch
Ground Truth
Convergence

Correspondence Vectors (red: x, blue: y)

Reconstruction of image A using patches from image B

Iteration 1

Iteration 3
Convergence

Correspondence Vectors (red: x, blue: y)

Reconstruction of image A using patches from image B
Convergence

Correspondence Vectors
(red: x, blue: y)

Reconstruction of image A using patches from image B
Image Completion

Input

Hole

Output (enlarged)
Image Completion

Input

Hole + constraints

Output (enlarged)
Image Completion
(without constraints)
Image Completion
(with constraints)

Input

Hole + constraints

Output (enlarged)
Retargeting / Reshuffling

Image retargeting [Simakov et al. ‘08]
(5 minutes per 250x200 input image)
Deformation Constraints
Deformation Constraints

Input

Partially retargeted
Deformation Constraints

Input

Partially retargeted
Deformation Constraints

Input

Partially retargeted
Deformation Constraints

Input

Partially retargeted
Deformation Constraints

Input

Partially retargeted
Line Constraints

Input

Improved
seam carving
[Rubinstein ‘08]

Our result
Region Constraints
Region Constraints

Original

Retargeted

With constraints
Reshuffling

Input

[Rubinstein ‘08]  [Wang ‘08]  PatchMatch
Reshuffling

Input

Enlarged

Reduced
Reshuffling

(demo played back at 5x real-time)
Local Scale

Boat marked by user

Scaled up, preserving texture

Tree marked by user

Scaled up, preserving texture
Limitations

- Poor convergence on pathological inputs
- Limits on scaling/rotation

Image A

Image B

Reconstruction of A from patches of B
Summary of Contributions
Future Work

Graphics – collages, video, new view synthesis, 3D

Vision – denoising, super-res, segmentation, object detection/recognition, irregularity detection

Algorithm – extend to other domains, GPU, …
Acknowledgements

• Sponsored in part by:
  – Adobe Systems
  – NSF grant IIS-0511965

• Thanks to:
  – SIGGRAPH paper reviewers
  – Tiggraph and Adobe CTL retreat reviewers
  – Jon Goldman, film footage from Kind of a Blur
  – Flickr users for Creative Commons imagery:
    Sevenbrane, Wili, Moi of Ra, Celie
Building marked by user

Tree marked by user

Scaled up, preserving texture

Scaled up, preserving texture