Computer Vision for Computer Graphics
(CVfCG)

Patch Correspondence

Max Planck Institute for Informatics (MPII)
Saarbrücken

Presenter: Nataliya Dedik
dediknat@gmail.com

Date: 28 May 2013

Outline

- Previous talks and connections
- Overview of patch correspondence problem
- Two papers
    - PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing (SIGGRAPH'09) [1]
    - NRDC: Non-Rigid Dense Correspondence with Applications for Image Enhancement (SIGGRAPH'11) [2]
- Summary, conclutions and connections
- Questions and discussions

Previous talks and connections

- Key ideas of previous talks:
  - recognize human actions at a distance
  - inserting new objects into existing photographs

- Connections to present works:
  - working with images
  - nearest-neighbour framework
  - correspondence problem, stitching
  - color value changes and optical flows, filtering

Overview

- Correspondence problem
  - occurrence of problem
  - basic methods
  - typical application examples

Source: http://en.wikipedia.org/wiki/Correspondence_problem

Outline

- Previous talk and connections (Illusion of Motion)
- Overview of patch Correspondence problem
- Two papers
  - PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing (SIGGRAPH'09) [1]
  - NRDC: Non-Rigid Dense Correspondence with Applications for Image Enhancement (SIGGRAPH'11) [2]
- Summary, conclutions and connections
- Questions and discussions

Problem description

- Digital image techniques
  - Image retargeting
  - Image completion
  - Image reshuffling

- Essential: user interaction
  - Flexibility
  - Quickness

- Most approaches meet only one of criterias

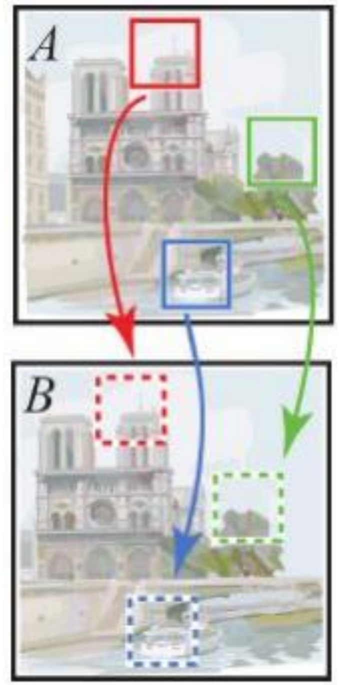
(a) input (b) result

(a) input (b) hole and guides (c) completion result

(a) input (b) result

Nearest-Neighbor Field (NNF)

- Repeated search for similar patches – core element
- Key observations:
  - Dimensionality of offset space
    - – patch offsets vs patches
  - Natural structure of images
    - – neighboring pixels have coherent matches
  - The law of large numbers
    - – large number of random sampling will yield good guesses



PatchMatch: previous related works

- Texture synthesis and completion
  - Non-parametric texture synthesis
  - Completion problem as a global optimization
  - Patch optimization
  - Parallel update schemes
  - Global formulation using Loopy Belief Propagation

- Relatively slow
- Only on small images

PatchMatch: previous related works

- Nearest neighbor search methods
  - Local propagation technique
    - Ashikhmin [2001]
  - k-coherence technique
    - Tong et al. 2002
  - Required to escape local minima
  - Compare to kd-trees

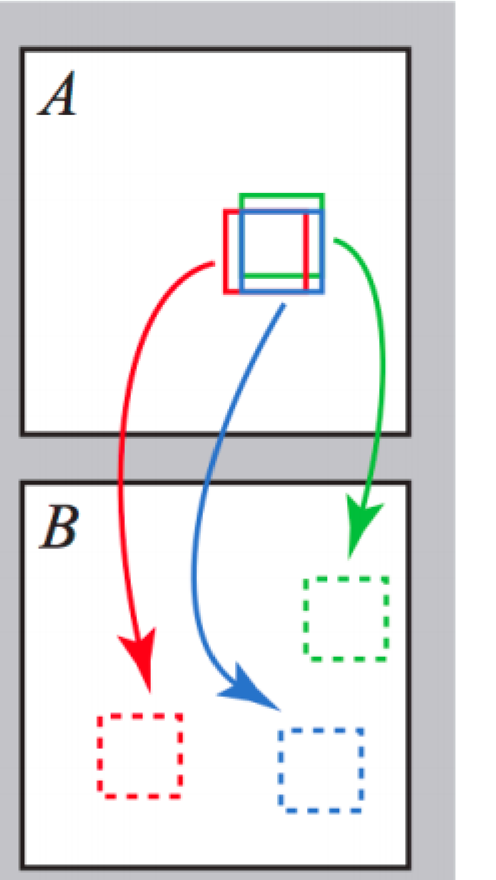
PatchMatch: previous related works

- Control and interactivity
  - user control by initializing the pixels with desired colors
    - Ashikhmin [2001]
  - “guiding layers”
    - Hertzmann et al. [2001]
  - structures that cross both inside and outside the missing region
    - [Sun et al. 2005]
  - deform image feature curves
    - Fang and Hart [2007]
  - interactive completion system
    - Pavic et al. [2006]

Approximate nearest-neighbor algorithm

- Three steps of algorithm:

1) Initialization
   - Each pixel is given a random patch offset as initialization.
     - Assign random values
     - Use prior information
   - Coarse-to-fine gradual resizing process
     - Initial guess upscaled from the previous level
     - sometimes get trapped in local minima
   - Random initialization

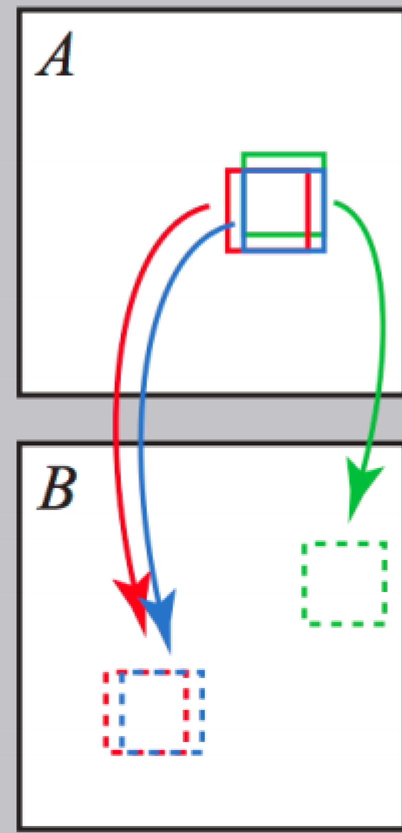


(a) Initialization

Approximate nearest-neighbor algorithm

2) Propagation

- Each pixels checks if the offsets from neighboring patches give a better matching patch.
- Adopt neighbor's patch offset.

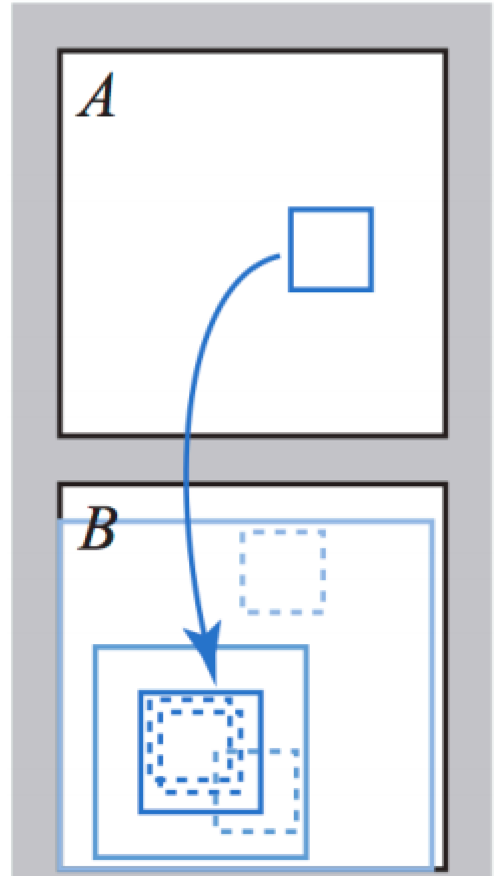


(b) Propagation

Approximate nearest-neighbor algorithm

3) Random search
   – searchs for better patch offsets
   – search radius - the size of the image, halved each time until it is 1.

- Criteria for halting – fixed number of times



(c) Search

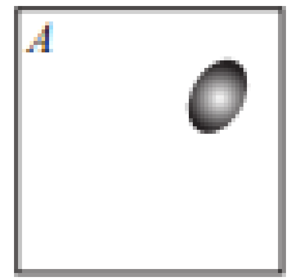
Overall algorithm

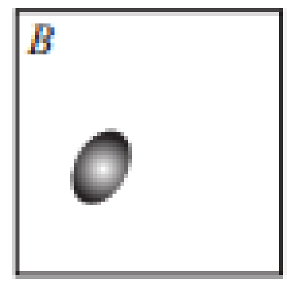
1. Initializing pixels with random patch offsets
2. Checking if neighbors have better patch offsets
3. Search in concentric radius around the current offset for better patch offsets
4. Go to Step 2 until converge

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

Analysis for a synthetic example

- Analyzing the convergence to exact NNF
- Challenging synthethic test case
- Distinctive region R will be correct via propagation
- Worst case for matching





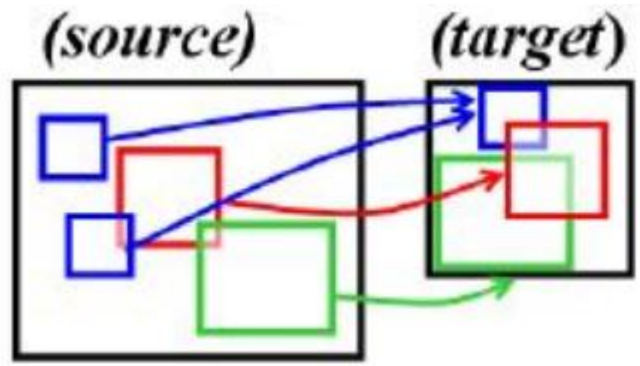
Analysis for real-world images

- Error analysis
  - Inputs and outputs of our editing operations
  - Stereo pairs and consecutive video frames
  - Images from the same class in the Caltech-256 dataset
  - Pairs of unrelated images
- Faster than kd-tree
- Uses less memory

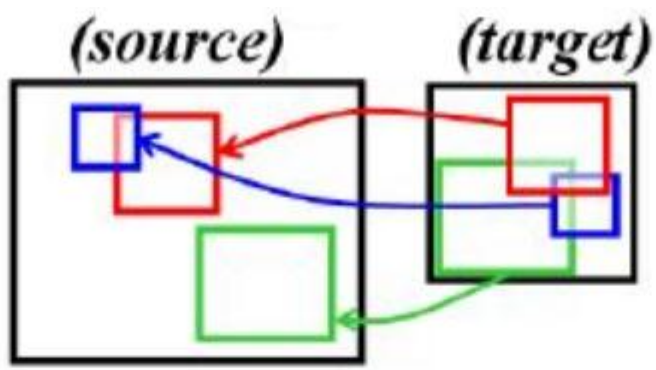
	Time [s]		Memory [MB]	
Megapixels	Ours	*kd*-tree	Ours	*kd*-tree
0.1	0.68	15.2	1.7	33.9
0.2	1.54	37.2	3.4	68.9
0.35	2.65	87.7	5.6	118.3

Editing tools

- Bidirectional similarity synthesis approach
  - Completeness



  - Coherence

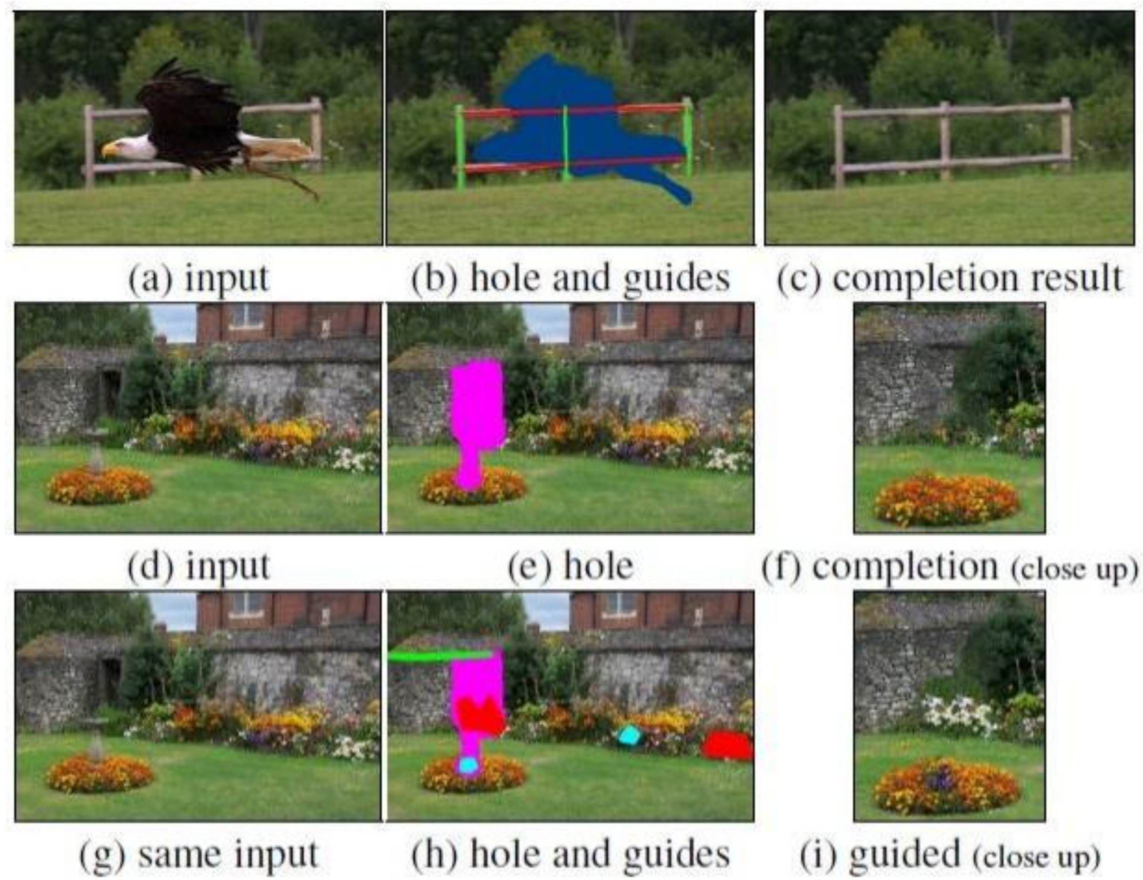


Editing tools

- Image collages
- Automatic cropping
- Reshuffling
- Importance masks
- Define constraints
- “copy and paste”
- Scaled uniformly or non-uniformly

Search space constraints

- Image completion is challenging
  - Inconsistencies
  - Boundaries
  - Limiting the search space



Deformation constraints

- For user to mark semantically important regions

(a) original (b) retargeted (c) with constraints

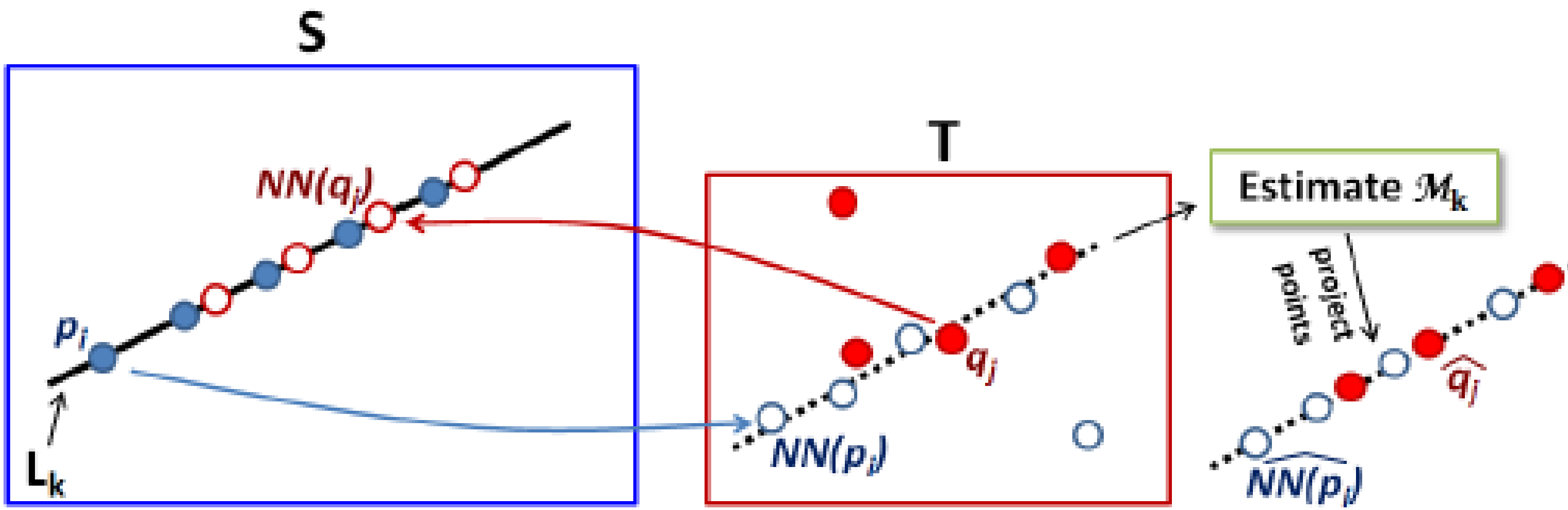
Figure 6: *Free lines and uniform scale constraints.*

(a) original (b) retargeted (c) with constraints

Figure 5: *Model constraints.*

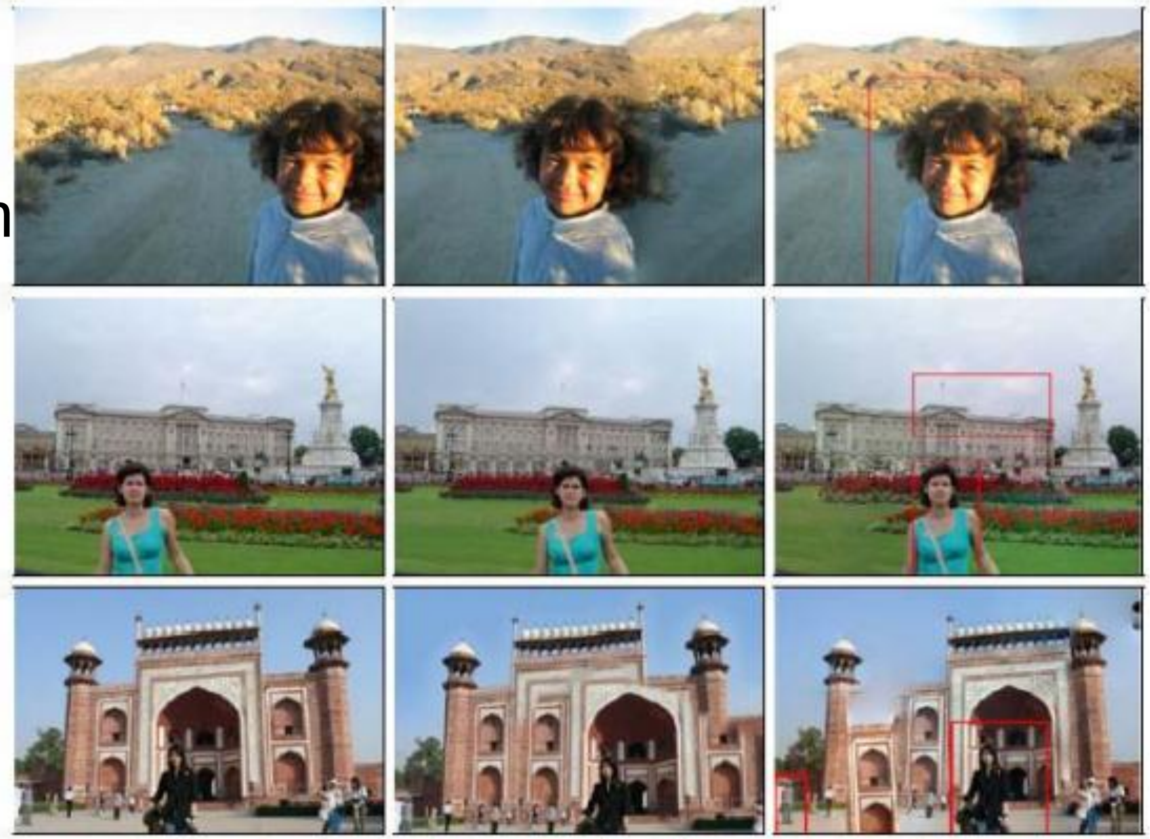
Deformation constraints

- Model constraints



Hard constraints (reshuffling)

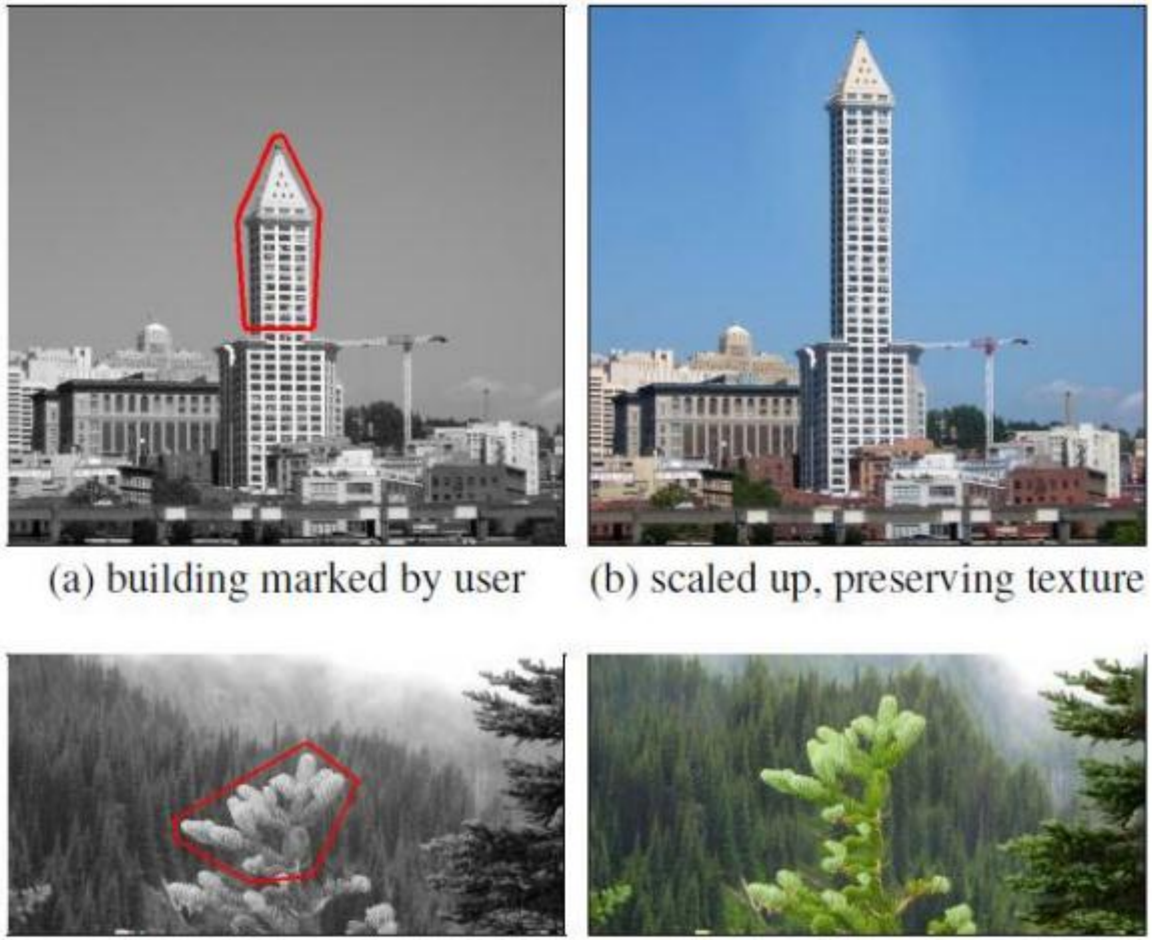
- Fixing the NN fields
- After each iteration correct offsets
- Three options to user
  - Swap
  - Interpolate
  - Clone



(a) input (b) our reshuffling (c) Patch Transform

Figure 8: *Examples of reshuffling.*

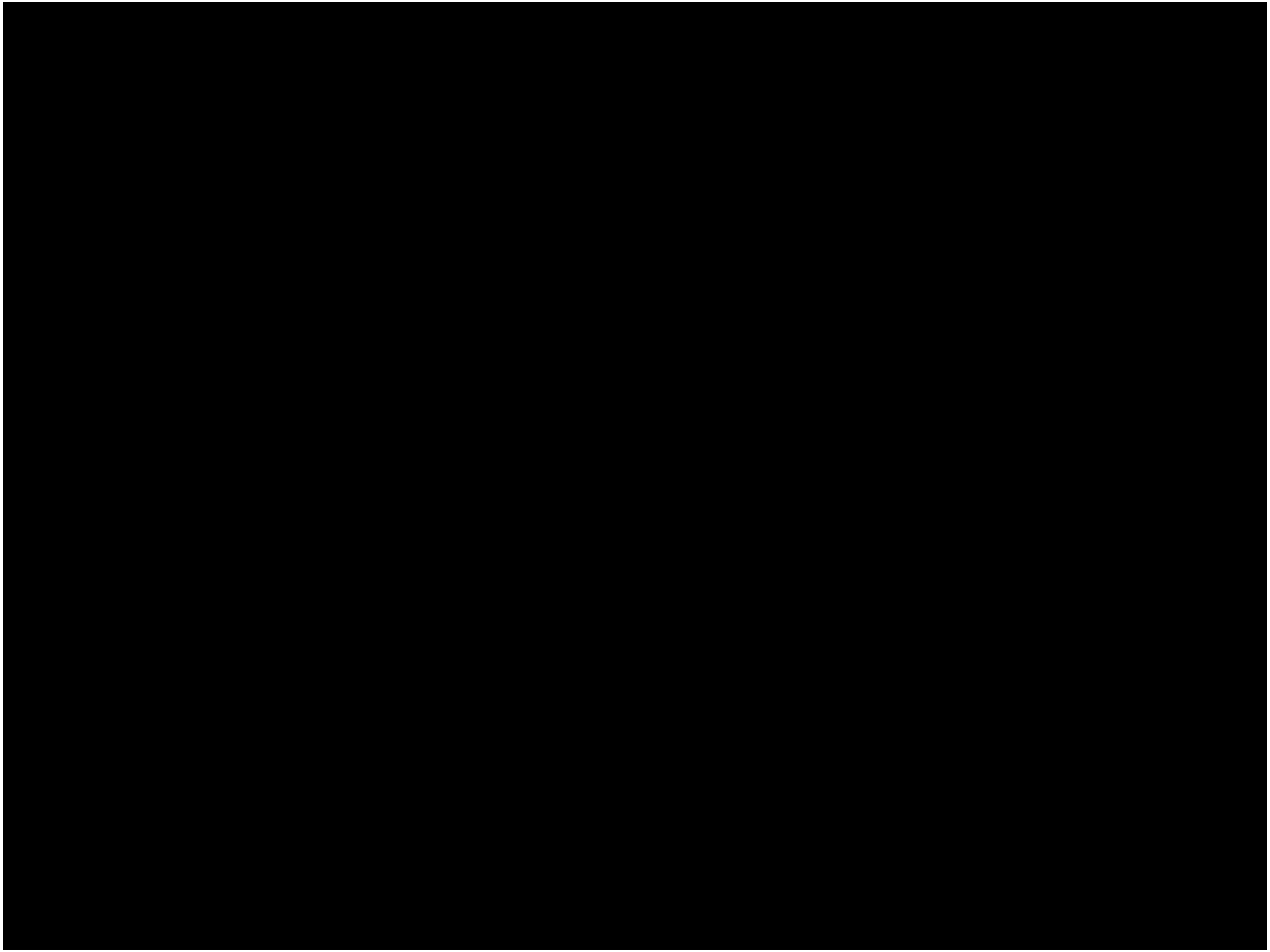
Local structural scaling



(a) building marked by user (b) scaled up, preserving texture

(c) bush marked by user (d) scaled up, preserving texture.

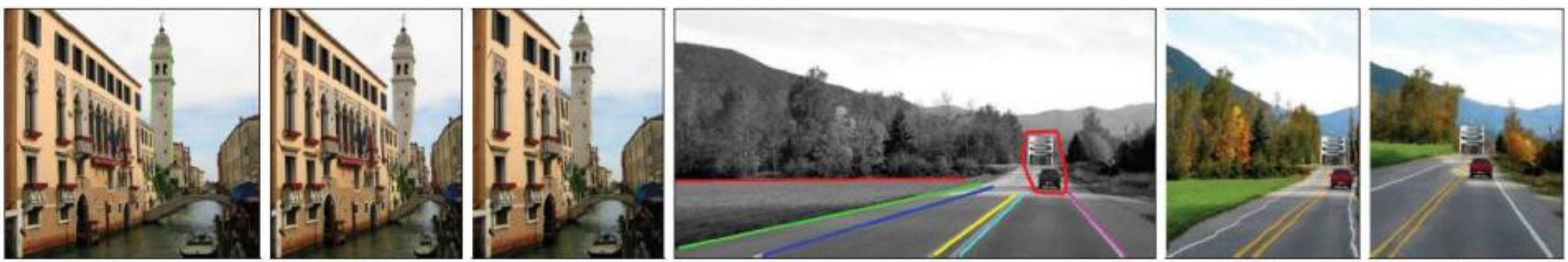
Figure 9: *Examples using local scale tool.*



Results

(a) (b) (c) (d) (e)

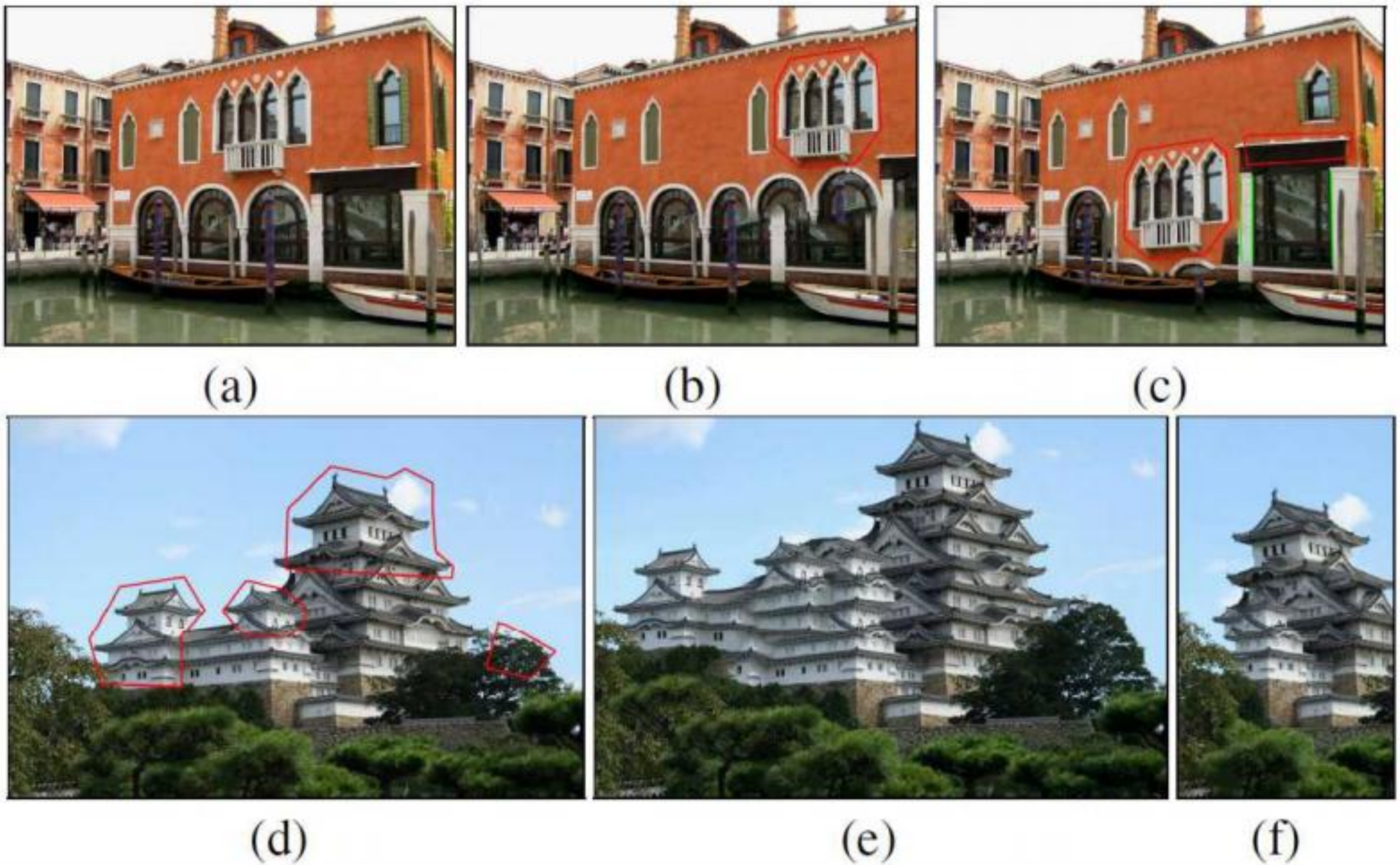
Figure 10: *Retargeting. From left: (a) Input image, (b) [Rubinstein et al. 2008], (c) [Wang et al. 2008], (d) Our constraints, (e) Our result.*



(a) (b) (c) (a) (b) (c)

Figure 11: *Retargeting: (a) Input image, annotated with constraints, (b) [Rubinstein et al. 2008], (c) Our output.*

Results



Pros and Cons

+

- Optimize an energy function without neighborhood term.
- Has no explicit generative model, but uses coherency in the data
- Sufficient for practical synthesis applications
- Avoids the expensive computations

-

- Poor convergence properties
- “Ghosting” or “feathering” artifacts

Future work

- k nearest neighbors – may allow the k-coherence strategy
- Optimal random sampling pattern and halting criteria - functions of the inputs

  -> exploring these tradeoffs:

  - New applications
  - Additional speed gains
- For videos
- For collages
- On 3D geometry, 4D animation or volumetric simulation sequences
- Object detection and tracking

Outline

- Previous talk and connections (Illusion of Motion)
- Overview of patch Correspondence problem
- Two papers
    - – PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing (SIGGRAPH'09) [1]
    - – NRDC: Non-Rigid Dense Correspondence with Applications for Image Enhancement (SIGGRAPH'11) [2]
- Summary, conclutions and connections
- Questions and discussions

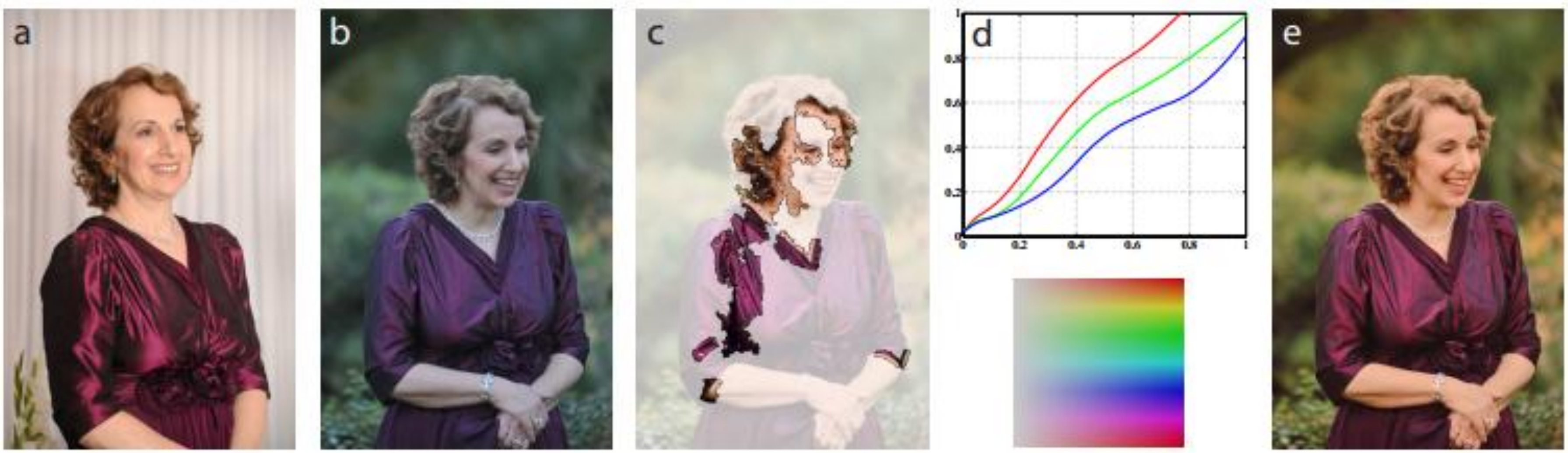
Key ideas

- Correspondence methods for two scenarios:
  - Close in time/view point
  - Difference in view point

- Third scenario:
  - Share common content
  - Factors of difference

Key ideas

- Color transfer: reference -> source
- Automatically recover a set of dence correspondances
- Global non-linear parametric color transformation model



- Dense and robust

Previous related work

- Correspondance:
- Initial Correspondence methods
  - Stereo matching
  - Optical flow
  - Image alignment
    - [Lukas and Kanade 1981]
- Features are robust to:
  - typical appearance variations
  - 3D transformations
- Geometric filtering steps
  - Less effective

[Pitié et al. 2007] SIFT Our result

Previous related work

- Large-displacement optical flow
- Non-rigid matching of different scenes

}
- Impressive results
- Not robust to changes

- Family of methods: start with feature matches -> “densify”
  - Objects with similar appearances
  - Scale not well

- Generalized PatchMatch (GPM)
  - Coarse-to-fine scheme
  - Iterative tonal
  - Color correction
  - Aggregation
  - Local narrowing

Correspondence algorithm

- Coarse-to-fine algorithm
  - nearest-neighbor search
  - region aggregation
  - color transform fitting
  - search range adjustment

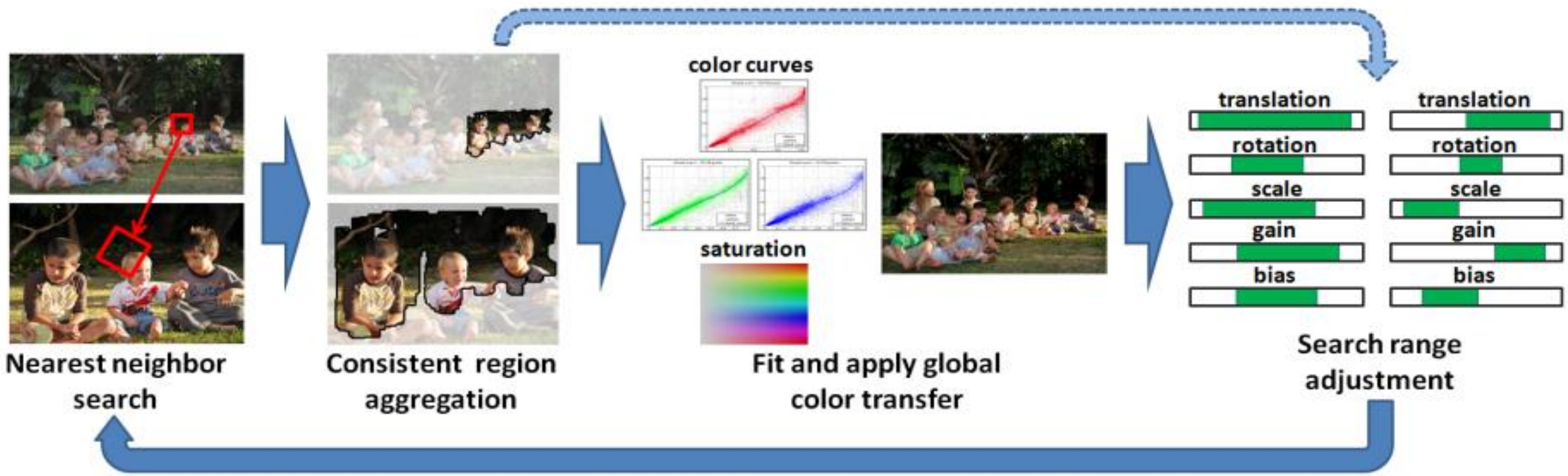


Figure 3: *The four steps of our correspondence algorithm - these are repeated several iterations at multiple scales.*

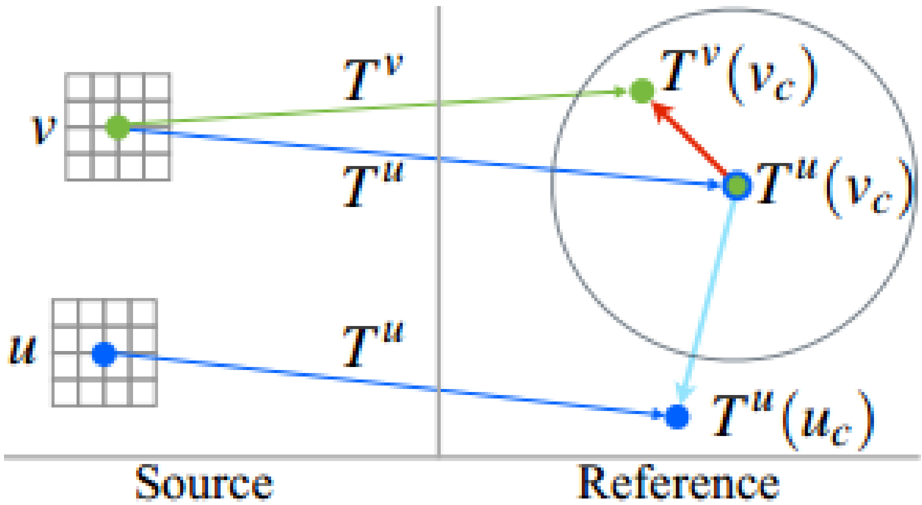
Correspondence algorithm

- Nearest-neighbor search - NNF from source to reference

- We use:
  - Small overlapping patches
  - Generalized PatchMatch extended to support
    - Robust color transformations
    - Sub-pixel translation
    - Gaussian weighted mean and variance
    - Mipmaps

Correspondence algorithm

- Aggregating consistent regions
- We use:
  - Consistency criterion -> Accept regions according to coherence error
  - Define adjacent patches as consistent -> Normalize -> Consistency error formula -> Compute the connected components



Correspondence algorithm

- Global color mapping
- Purposes:
  - improve the performance of the correspondence algorithm
  - produce the final result
- We use:
  - Three monotonic curves
  - Piecewise cubic spline with 7 breaks
  - Quadratic programming
  - Handle saturation changes -> matrix->project pixel colors and optimize for the scale factors

Correspondence algorithm

- Search constraints
- 8 additional degrees of freedom
- Problem: incorrect matches
- Overcome: limit the search range of transformations
  - Consistency criterion
  - Narrow the search range

Evaluation

- Time depends on:
  - Interpolation methods
  - Number of GPM iterations

- Results:
  - Immediate feedback
  - Improvement of correspondance

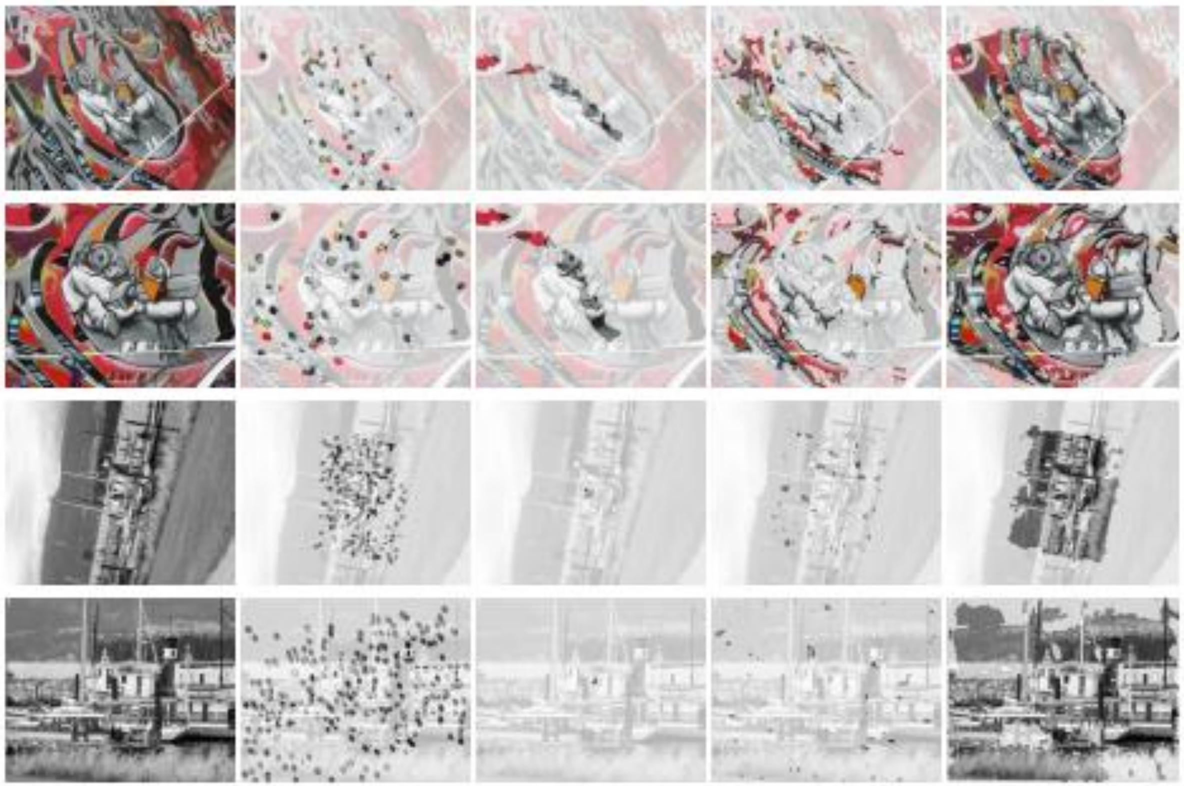
Evaluation: Correspondence evaluation

- Compare our algorithm with
  - SIFT-Flow and Generalized PatchMatch
  - sparse SIFT correspondence

(a) Inputs (b) SIFT (c) GPM (d) Our

Evaluation: Correspondence evaluation

- Evaluation the accuracy of correspondence quantitatively



(a) Inputs (b) SIFT (c) SIFT-Flow (d) GPM (e) Our

Evaluation: Correspondence evaluation

- Comparison with the Co-recognition approach

(a) Inputs (b) Co-recognition (c) Our

Evaluation: Global color transfer evaluation

- Comparison to the state-of-the-art method

(a) Reference (b) Source (c) Pitié *et al.* (d) Our

Evaluation: Limitations and future work

- Very large smooth regions
- Object appears over different background
- Two or more very different color models
- Will handle only saturation/desaturation

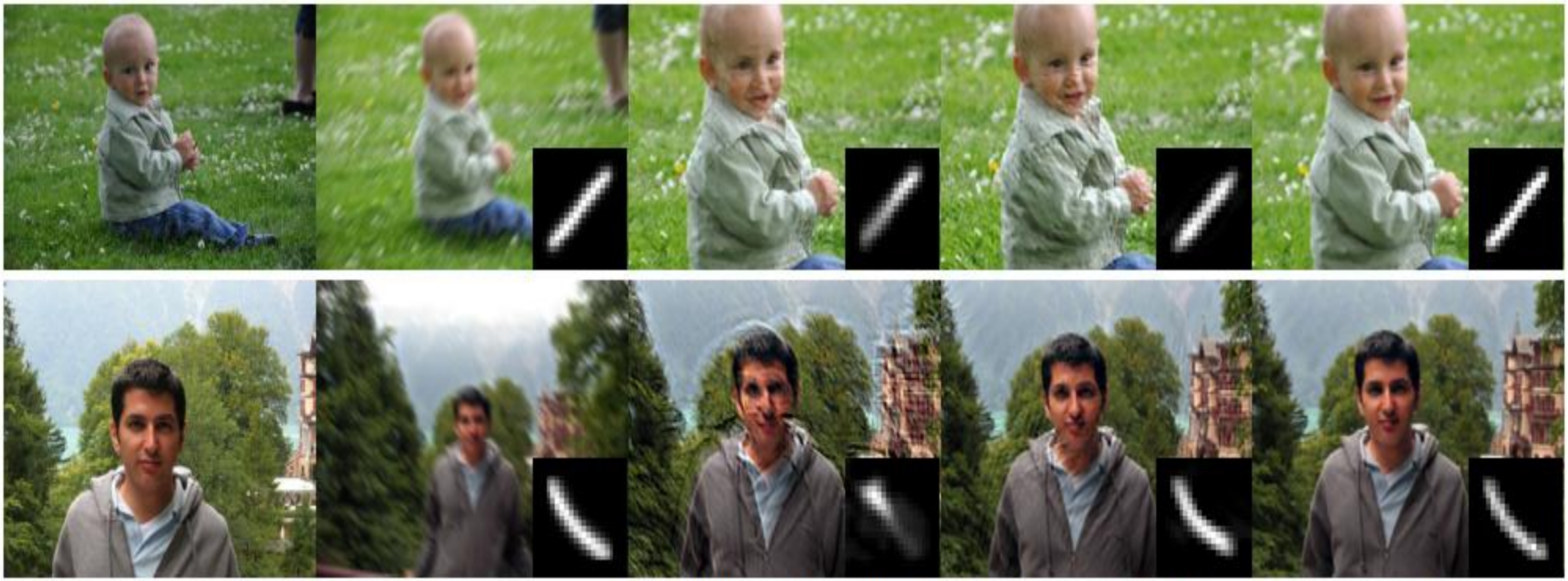
Applications

- Local color transfer

(a) Reference (b) Source (c) Matches (d) Our global (e) Our local (f) Zoom-in (global) (g) Zoom-in (local)

Applications

- Deblurring



(a) Sharp example (b) Blurry (c) [Cho and Lee 2009] (d) [Levin et al. 2011] (e) Our

Applications

- Mask transfer


(a) Reference (b) Input mask (c) Source (d) Trimap (e) Output mask

Future work

- For videos
- On 3D geometry,
- 4D animation or volumetric simulation sequences
- Computer graphics and vision applications that rely on correspondence methods.

Outline

- Previous talk and connections (Illusion of Motion)
- Overview of patch Correspondence problem
- Two papers
  - PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing (SIGGRAPH'09) [1]
  - NRDC: Non-Rigid Dense Correspondence with Applications for Image Enhancement (SIGGRAPH'11) [2]
- Summary, conclutions and connections
- Questions and discussions

Summary: PatchMatch

- Image editing tools using a new randomized algorithm for quickly finding approximate nearest-neighbor matches between image patches.
  - Image retargeting
  - Completion
  - Reshuffling
- Interactivity is essential

Summary: NRDC

- Recovering reliable local sets of dense correspondences between two images with some shared content
  - Dense local matching
  - Robustness to outliers
- Applications:
  - Adjusting the tonal characteristics to match a reference image
  - Mask transferring
  - Kernel estimation for image deblurring

Connections

- Connect to correspondence problem
- PatchMatch algorithm
- Coarse-to-fine scheme
- Tools for image editing
  - Retargeting, completion, reshuffling
  - Tonal characteristics, transfer mask, deblurring
- Colors and flows
- Can be used for videos editing in future

Conclusion

- Acknowledgements: Thanks to James Tompkin

- References:

  - [1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, Dan B Goldman PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. ACM Transactions on Graphics (Proc. SIGGRAPH), August 2009.
  - [2] Yoav HaCohen, Eli Shechtman, Dan B. Goldman, Dani Lischinski. NRDC: Non-Rigid Dense Correspondence with Applications for Image Enhancement. ACM Transactions on Graphics (Proc. SIGGRAPH), August 2011.

Thank you for your attention!

- Questions and discussion