Drag and Drop Pasting

Jiaya Jia, Jian Sun, Chi-Keung Tang, Heung-Yeung Shum

The Chinese University of Hong Kong
Microsoft Research Asia
The Hong Kong University of Science and Technology

Presented By

Bhaskar Kishore

10/8/07

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Introduction

- Problem
  - For Poisson Image Editing to work well user must carefully draw boundary on source image
  - Poisson Image Editing may generate unnatural blurring artifacts

Introduction

- Proposed
  - A new objective function to compute an optimized boundary
  - A blend guidance field to integrate an alpha matte into the Poisson equation to preserve fractional boundary

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Related Work

- Poisson Image Editing [Perez et al. 2003]
- Image Stitching in the Gradient Domain [Levin et al. 2004]
- Interactive Digital Photomontage [Agarwala et al. 2004]
- Multi-resolution spline technique [Burt & Adelson 1983]

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Poisson Image Editing

- Minimizing Problem [Perez et al. 2003]
  - $$\min_f \int_{p \in \Omega_0} |\nabla f - v|^2 dp \text{ with } f|_{\partial\Omega_0} = f_t|_{\partial\Omega_0}, \qquad (1)$$

    where $v = \nabla fs$
  - Denote f' = f – fs, equation (1) becomes
  - $$\min_{f'} \int_{p \in \Omega_0} |\nabla f'|^2 dp \text{ with } f'|_{\partial\Omega_0} = (f_t - f_s)|_{\partial\Omega_0}. \qquad (2)$$
  - The associated Laplace equation is :

    $$\Delta f' = 0 \text{ with } f'|_{\partial\Omega_0} = (f_t - f_s)|_{\partial\Omega_0}, \qquad (3)$$

    where $\Delta = (\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})$ is the Laplacian operator

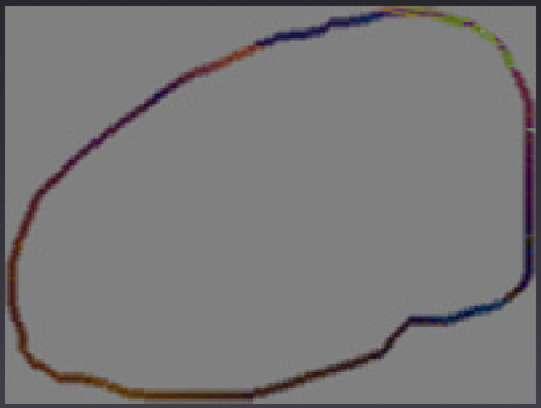
Poisson Image Editing

$$\min_{f'} \int_{p \in \Omega_0} |\nabla f'|^2 dp \text{ with } f'|_{\partial \Omega_0} = (f_t - f_s)|_{\partial \Omega_0}. \quad (2)$$

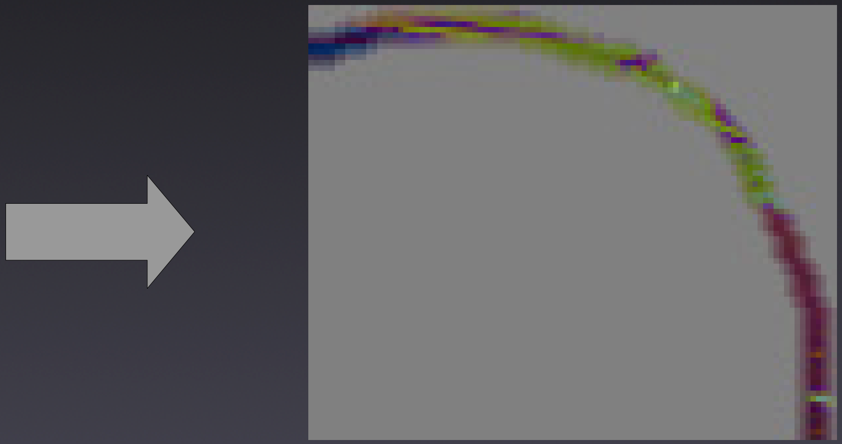
- Important Property of equation 2
  - Variational energy $\Omega_0 \; |\nabla f|^2$ will approach zero if and only if all boundary pixels satisfy $(f_t - f_s)|_{\partial\Omega 0} = k$, where k is some constant value [Zwillinger 1997]
  - Important as less variation in color along the boundary produces better results in the compositing operation

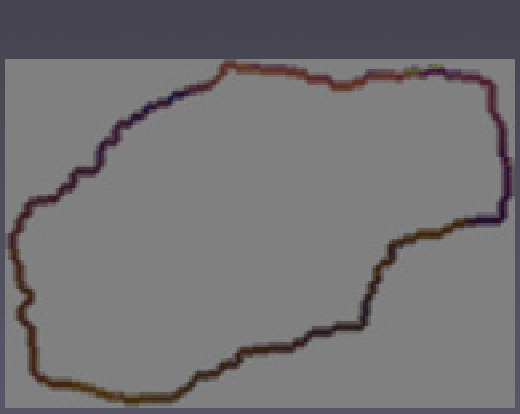
Poisson Image Editing

$$\min_{f'} \int_{p \in \Omega_0} |\nabla f'|^2 dp \text{ with } f'|_{\partial\Omega_0} = (f_t - f_s)|_{\partial\Omega_0}. \quad (2)$$

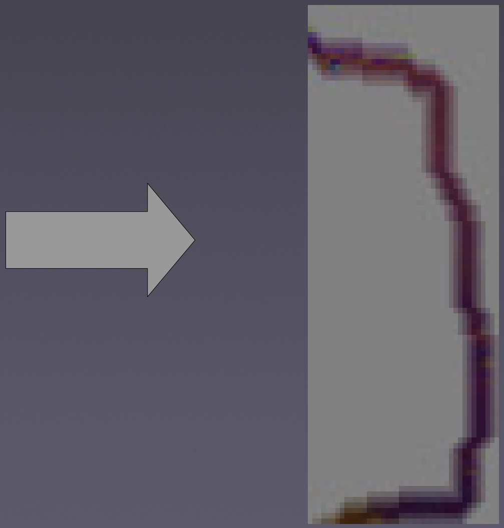


User drawn boundary





Optimal boundary

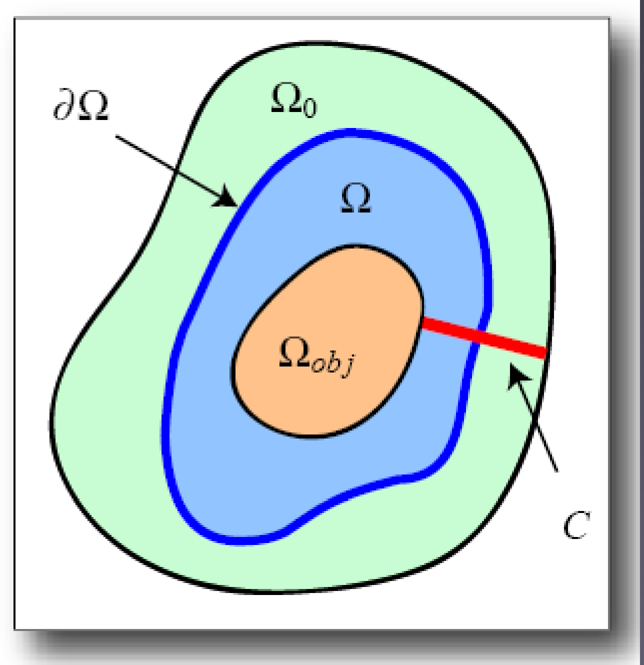


Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Before Boundary Energy Minimization

- Optimal Boundary
  - Where is this optimal boundary $\partial\Omega$?
  - Has to be inside the region of interest Ω_0
  - It also has to be outside the object of interest Ω_{obj}
  - Use GrabCut [Rother et al. 2004] to automatically compute Ω_{obj}



Boundary Energy Minimization

- To reduce the color variance along the boundary the following objective function is minimized :

$$E(\partial\Omega, k) = \sum_{p \in \partial\Omega} ((f_t(p) - f_s(p)) - k)^2, \text{ s.t. } \Omega_{obj} \subset \Omega \subset \Omega_0, \quad (4)$$

  - Where
    - f(p) is a ternary set in {r,g,b} space
    - f(p) – f(q) is computed as L2 Norm

Boundary Energy Minimization

- Iterative Optimization

1. Initialize Ω as Ω0
2. Take derivative of equation 4 and equate to zero. Solve for K.

$$\frac{\partial E(\partial\Omega, k)}{\partial k} = 0$$

$$\Leftrightarrow \quad k = \frac{1}{|\partial\Omega|} \sum_{p \in \partial\Omega} (f_t(p) - f_s(p)), \qquad (5)$$

where $|\partial\Omega|$ is length of the boundary $\partial\Omega$

3. Given the current K, optimize the boundary $\partial\Omega$.
4. Repeat steps 2 and 3 until the energy does not decrease in successive iterations

Boundary Energy Minimization

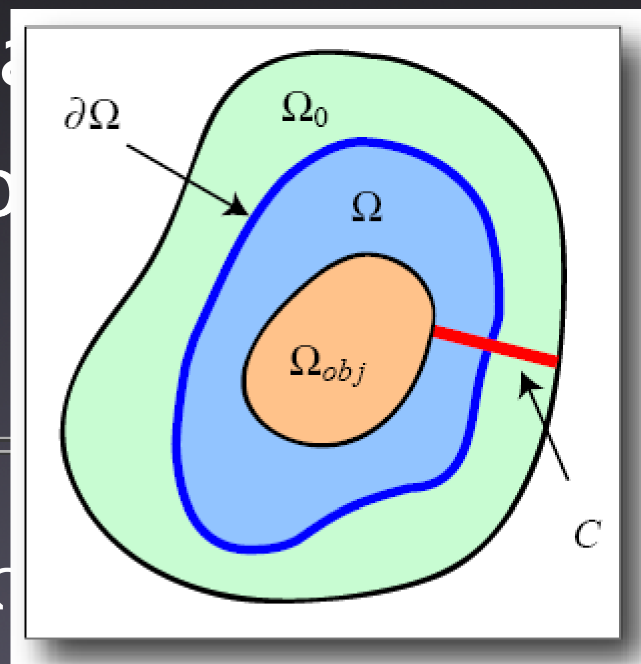
- Iterative Optimization

  1. Initialize Ω as Ω0

  2. Take deriva … n 4 and equate to zero. Solve fo …

$$\Leftrightarrow \quad k = \ \ldots \ f_s(p)), \qquad (5)$$

where |∂Ω … ary ∂Ω



  3. Given the current K, optimize the boundary ∂Ω.

  4. Repeat steps 2 and 3 until the energy does not decrease in successive iterations

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Shortest Closed Path

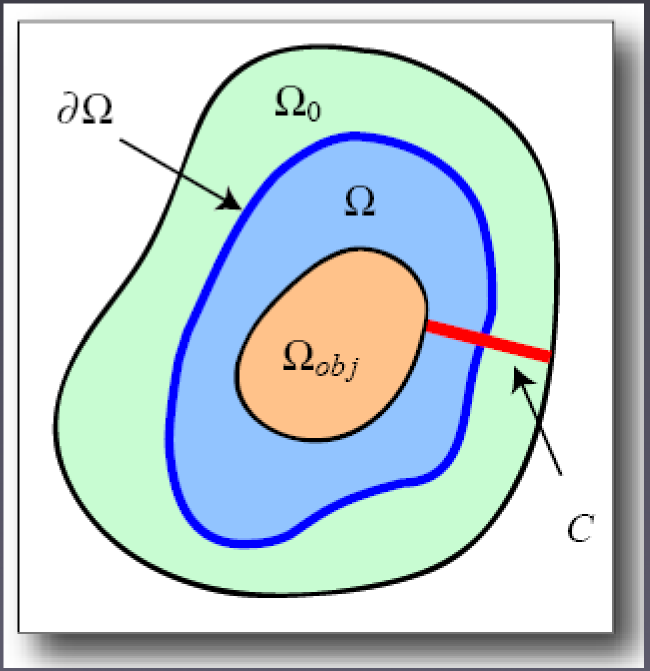
- Assume a graph G
  - Nodes in G are pixels within the band $\Omega - \Omega_{obj}$
  - Edges represent 4-connectivity relationships between neighboring pixels
  - Cost on each node is defined as

$$((f_t(p) - f_s(p)) - K)^2$$

  - Region Ω_{obj} can be considered as genus 0
  - Region $\Omega_0 \backslash \Omega_{obj}$ can be considered as genus 1

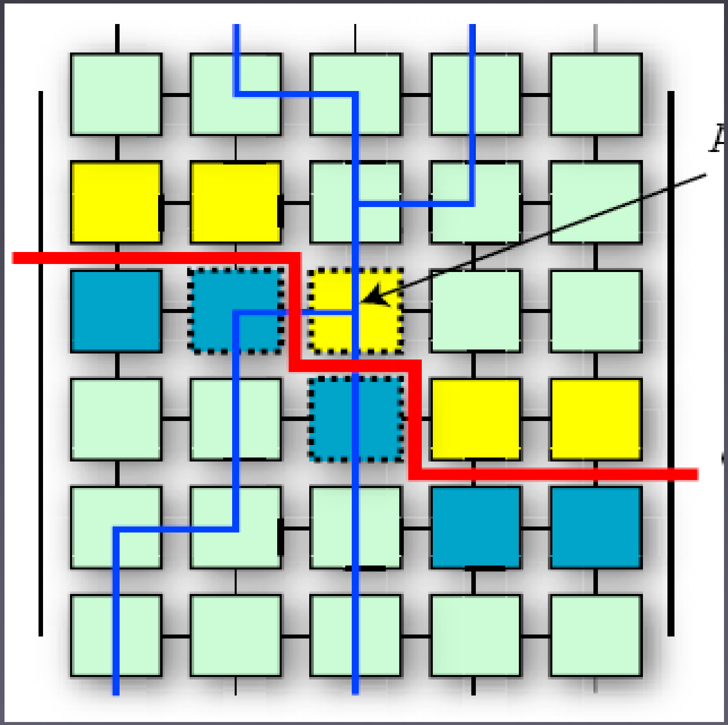
Shortest Closed Path

- Finding Shortest path is difficult as $\partial\Omega$ is a closed curve enclosing Ω_{obj}
- We therefore change region $\Omega_0 \backslash \Omega_{obj}$ from genus 1 to genus 0 by introducing cut C.



Shortest Closed Path

- Cut C
  - Compute shortest straight line segment among all pixel pairs for ∂Ωobj and ∂Ω0.
  - Benefits of this approach
    - Short length reduces probability that optimal boundary passes the cut more that once
    - Speeds up computation.



Shortest Closed Path

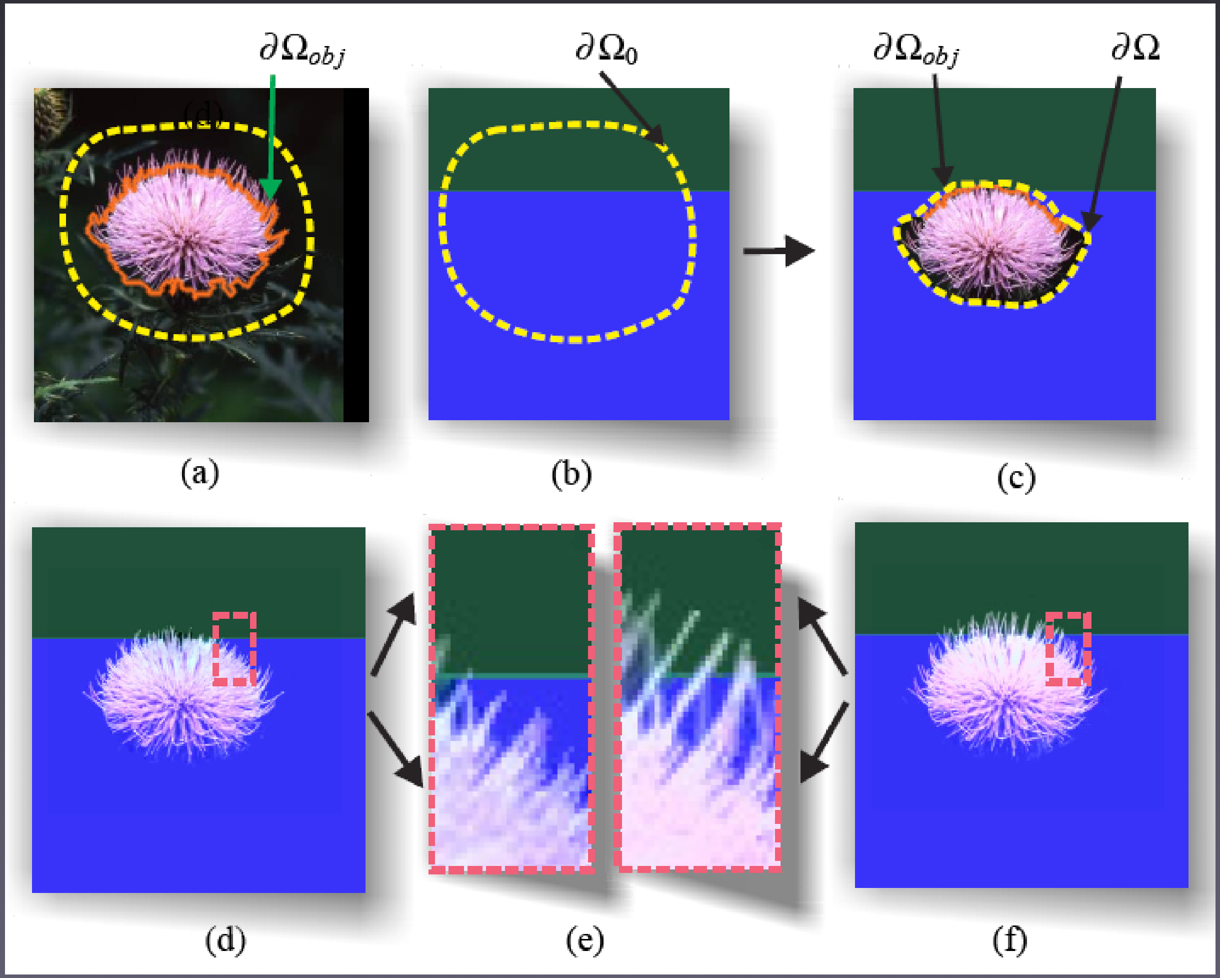
- Shortest Closed Path Algorithm
  - For each pixel P on one side of the cut (Marked Yellow) compute Shortest paths to all adjacent pixels on the other side (Shown in Blue).
    - Use 2D Dynamic programming [Dijkstra 1959; Mortensen and Barrent 1995] to compute paths.
    - O(N) where N is the number of pixels in the band.
  - Repeat process for all pixels on yellow side of cut C.
    - We obtain a set of Paths.
    - Optimized boundary $\partial\Omega$ is assigned to the path with globally minimum cost.
    - Assuming there are M yellow pixels, overall complexity is O(MN)

Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Fractional Boundary

- Problem is optimized boundary may intersect with an object with fractional boundary and break up subtle and fine details



Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

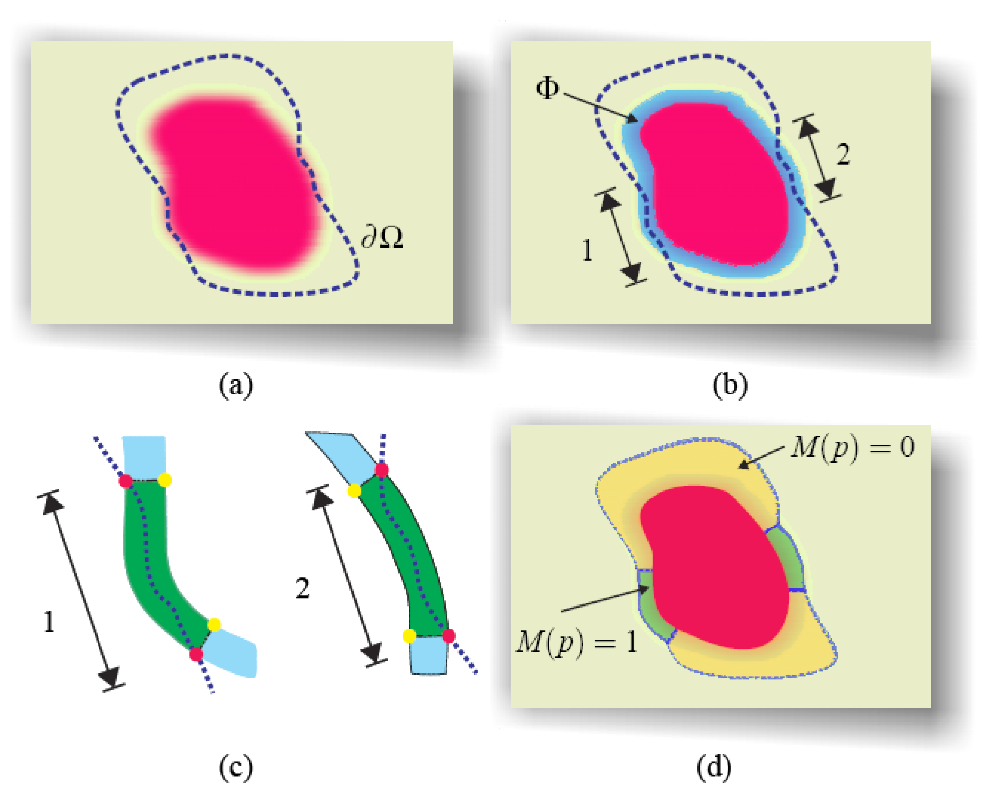
Blended Guidance Field

- Solution is to incorporate an alpha matt into the guidance field for Poisson equations
- To do this we must detect regions in which alpha blending should be computed
- Define binary coverage mask M to indicate where alpha blending should be applied.

Blended Guidance Field

- Denote Φ = { p |0 < α(p) < 1} as the fractional object boundary
- α is computed automatically within a few pixels surrounding Ωobj using coherence matting [Shum et al. 2004]

Φ is the shape of the narrow blue belt

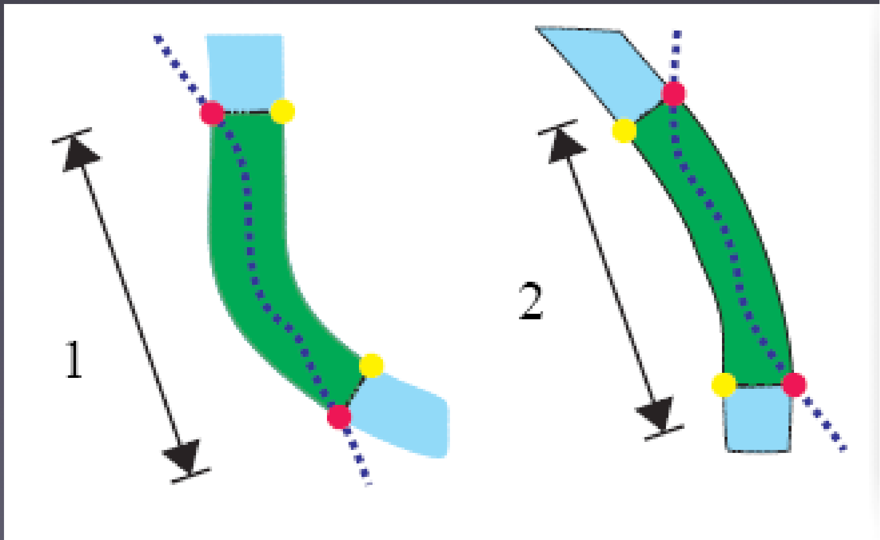


Blended Guidance Field

- Now the blended guidance field $v' = (v'_x, v'_y)$
- For each pixel $p = (x,y)$, $v'_x(x,y)$ is defined as

$$v'_x(x,y) = \begin{cases} \nabla_x f_s(x,y) & M(x,y) = M(x+1,y) = 0 \\ \nabla_x(\alpha f_s + (1-\alpha) f_t) & M(x,y) = M(x+1,y) = 1 \\ 0 & M(x,y) \neq M(x+1,y) \end{cases} \quad (6)$$

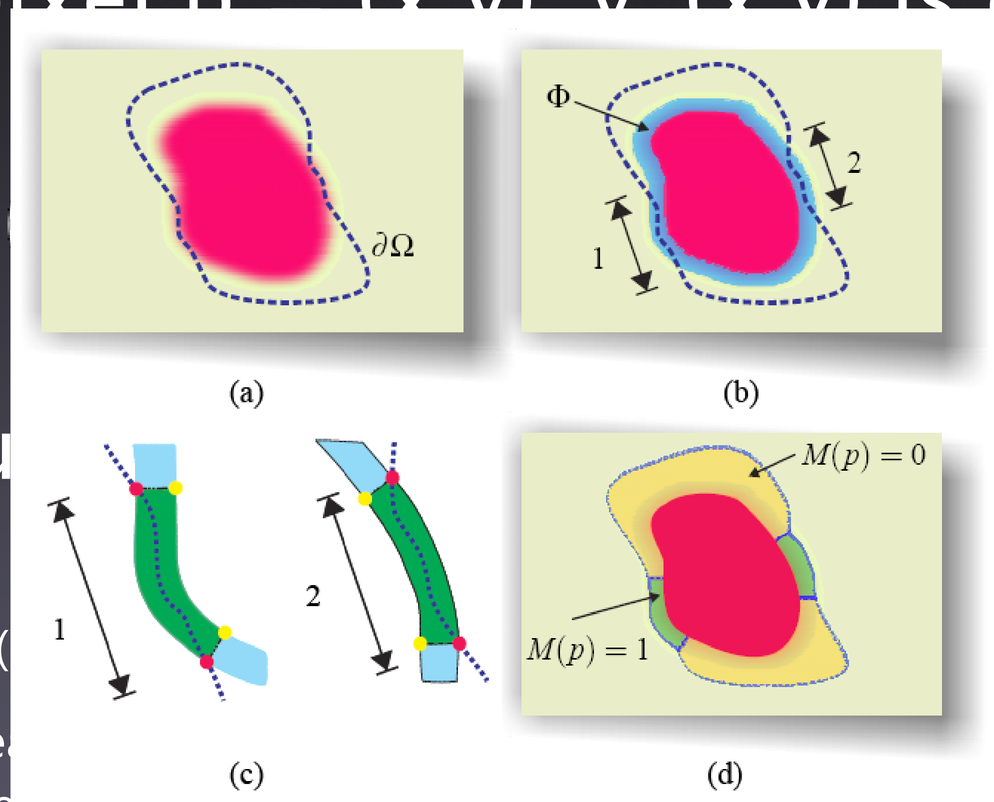
- To construct M
  - Compute head and tail intersections b/w $\partial\Omega$ and belt Φ (Red Dots)
  - Compute the nearest point on the other side of belt Φ . This gives us the green region in which blending must be applied.
  - Set $\{p|M(p) = 1\}$ in green region and $M(p) = 0$ in remaining pixels of p in $\Omega \cup \Phi$



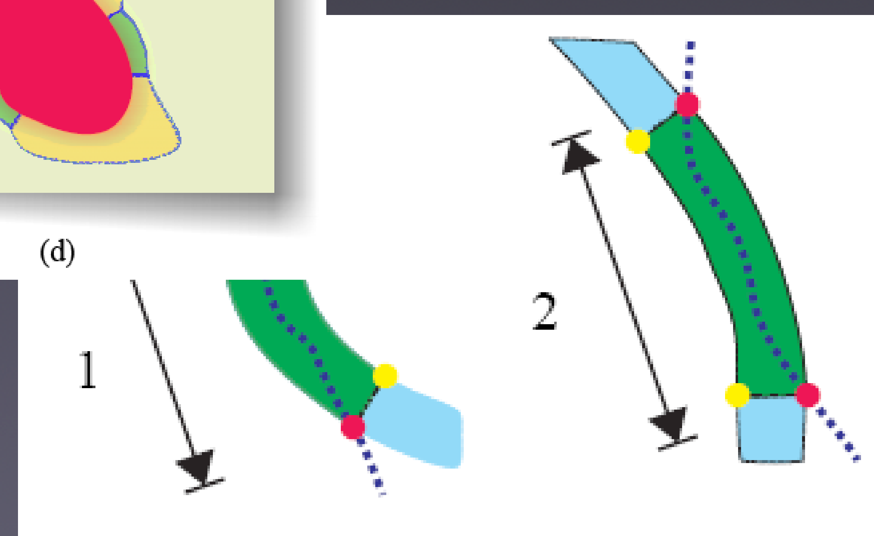
Blended Guidance Field

- Now the blended guidance field v' = (v'$_x$, v'$_y$)
- For each pixel p = (x,y), v'(x,y) is defined as

$$v'_x(x,y) = \begin{cases} \nabla_x \ldots & \ldots + 1, y) = 0 \\ \ldots & \ldots + 1, y) = 1 \\ \ldots & \ldots + 1, y) \end{cases} \quad (6)$$



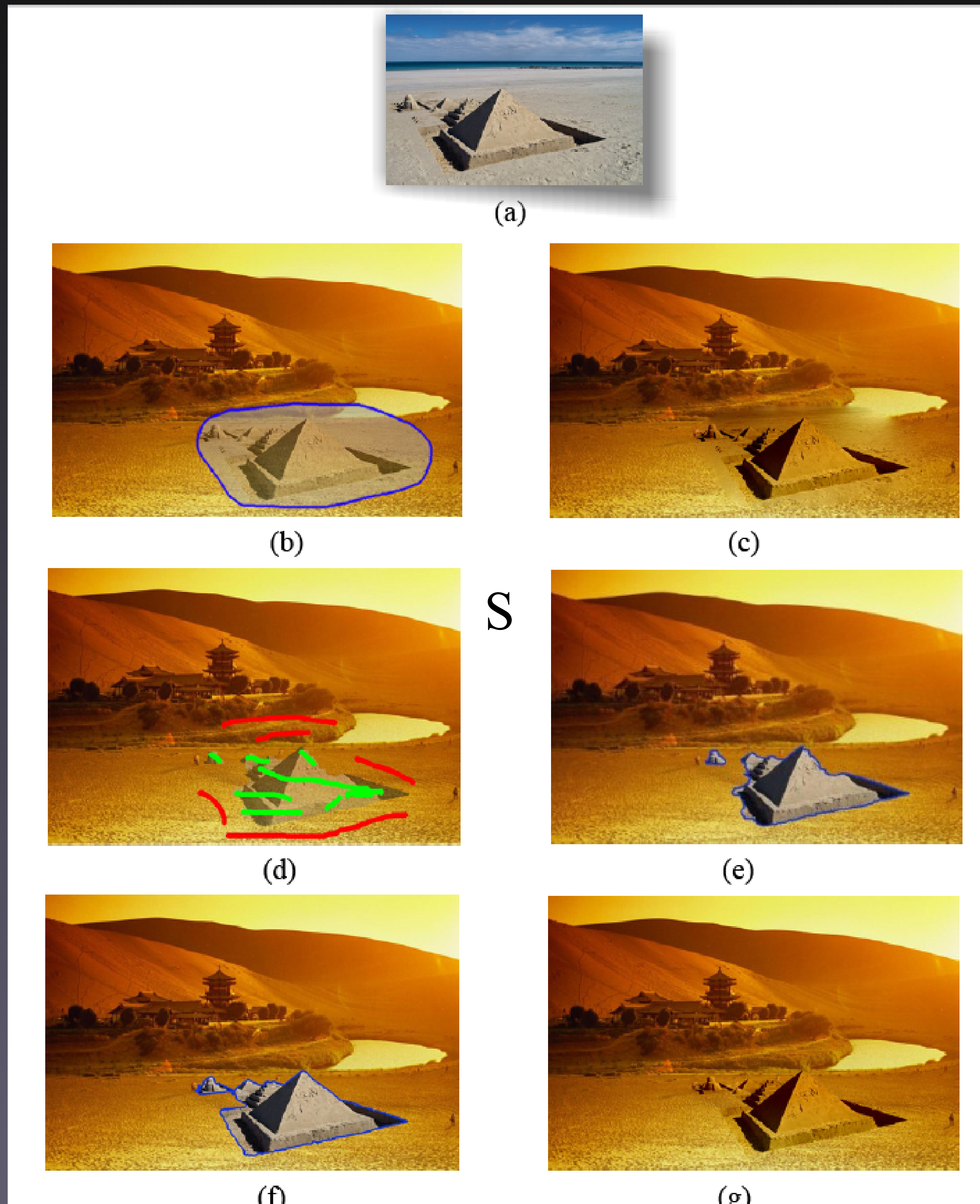
- To constru…
  - Compute head…
    ∂Ω and belt Φ (…
  - Compute the ne…
    belt Φ . This gives us the green region in which blending must be applied.
  - Set {p|M(p) = 1} in green region and M(p) = 0 in remaining pixels of p in Ω U Φ



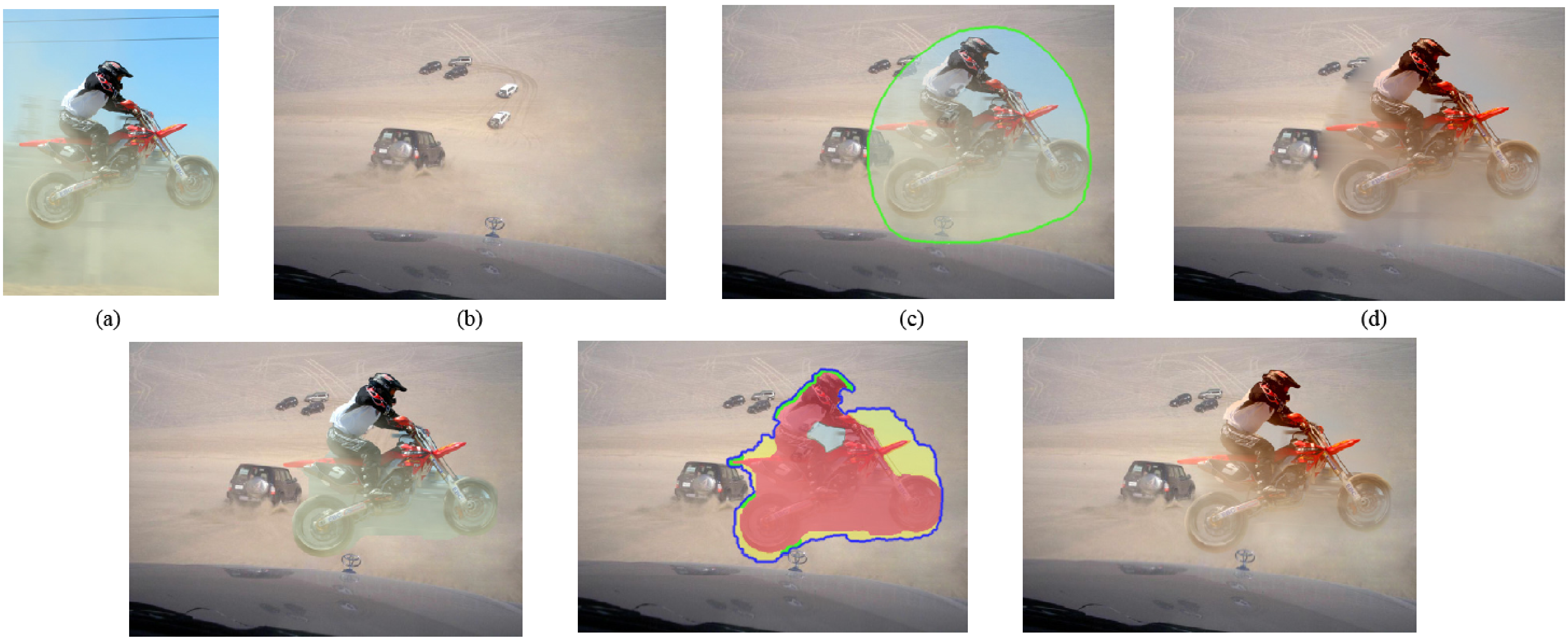
Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Results



Results



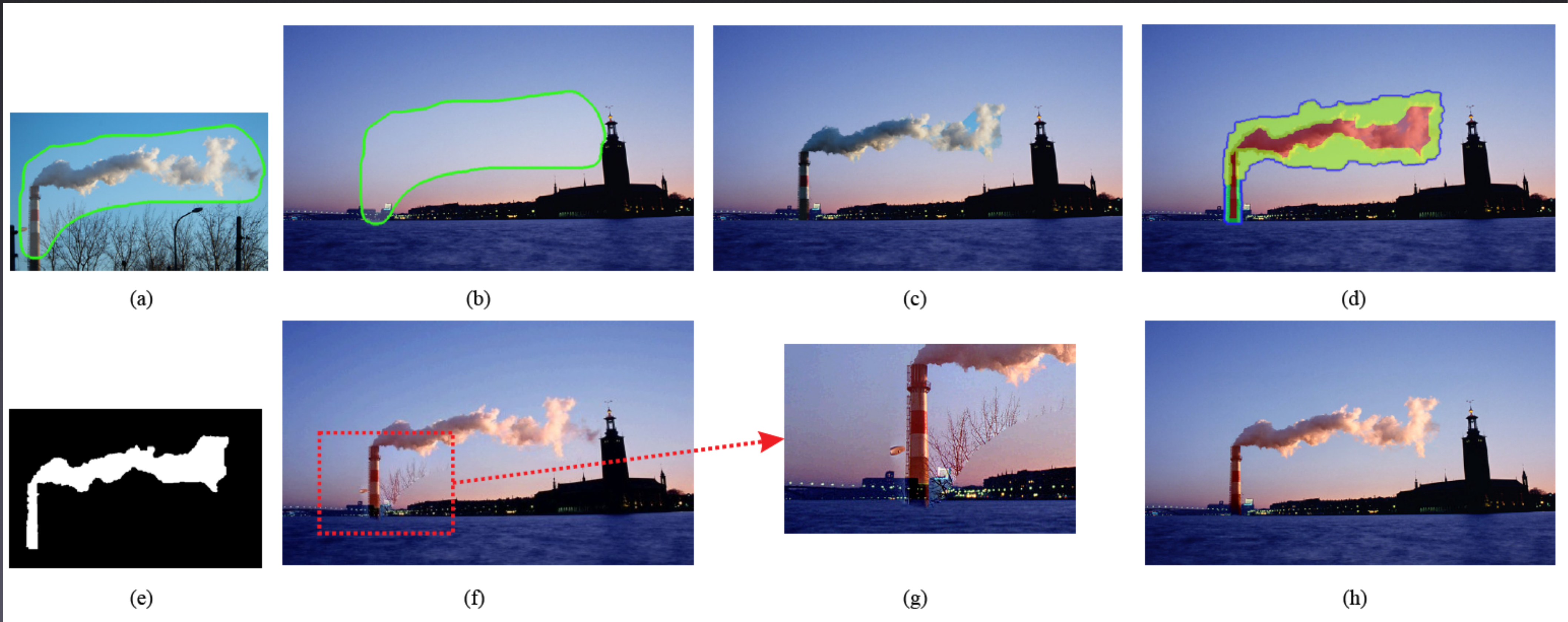
(a) (b) (c) (d)

(e) (f) (g)

Results

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

Results



Outline

- Introduction
- Related Work
- Optimal Boundary
  - Poisson Image Editing
  - Boundary Energy Minimization
  - Shortest Closed Path Algorithm
- Fractional Boundary
  - Blended Guidance Field
- Results
- Conclusions

Conclusions

- Proposed a user friendly approach to achieve seamless image composition without requiring careful initialization by user
- A system which is more practical and easy to use
- The approach preserves fractional boundary by introducing blended guidance field
- User interaction is only required if there is a large error in computing Ω_{obj}