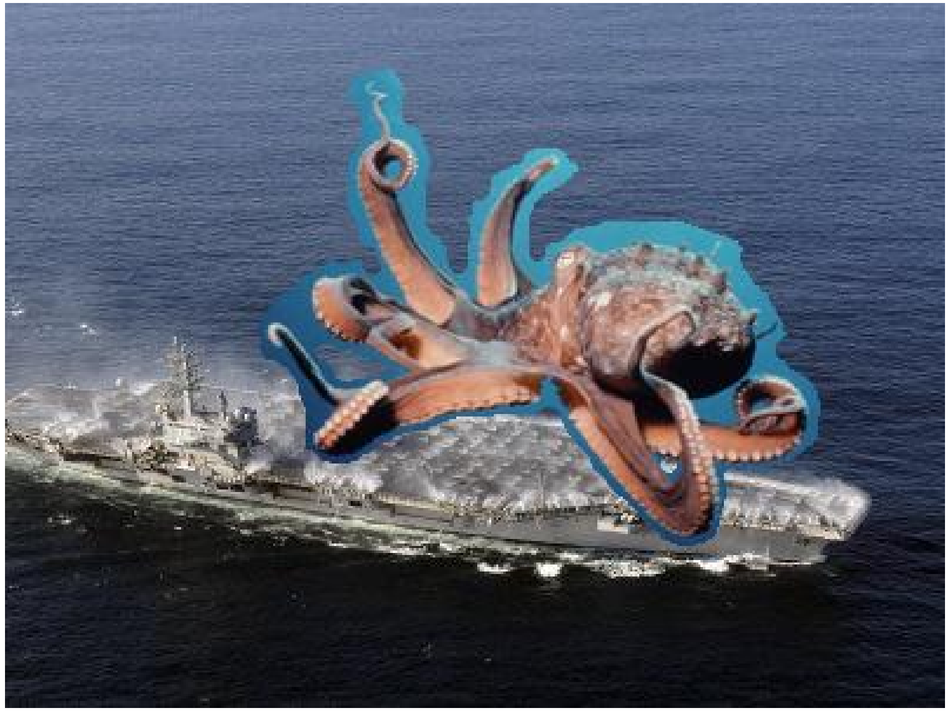
Poisson Image Editing

Yang Zhuoyu

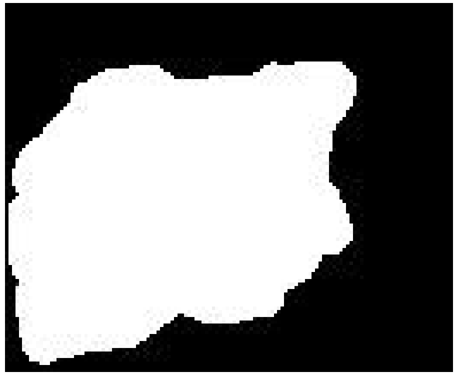
Goals

- Seamlessly importing (cloning) opaque or transparent source image regions into a target image.



Input Data

Output Data

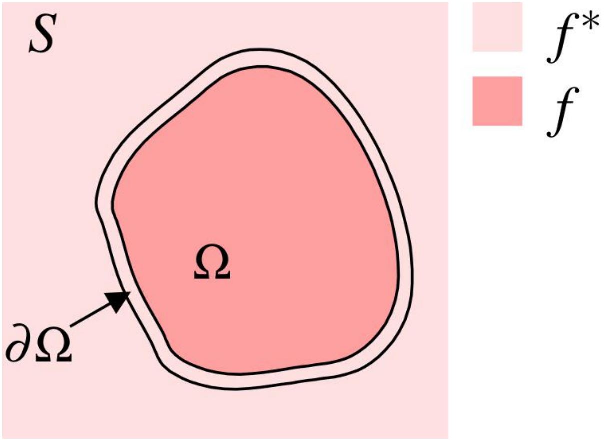


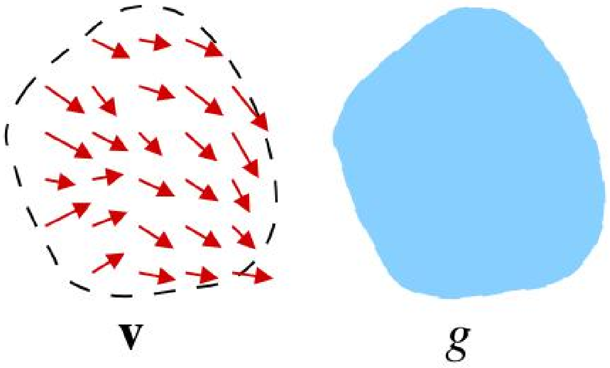
Mask

Source

Target

Guided interpolation notations





S, a closed subset of R^2 , be the image definition domain.

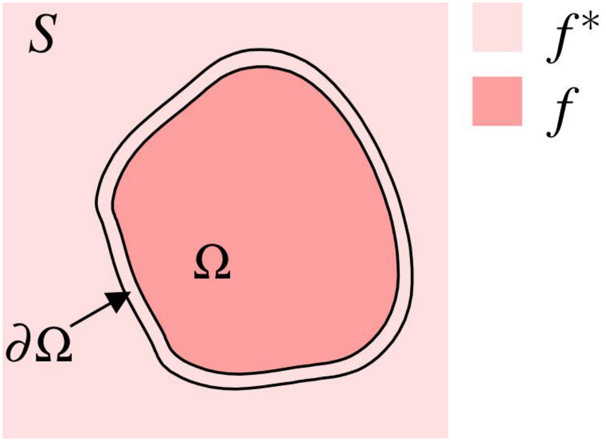
Ω, a closed subset of *S* with boundary *∂Ω*.

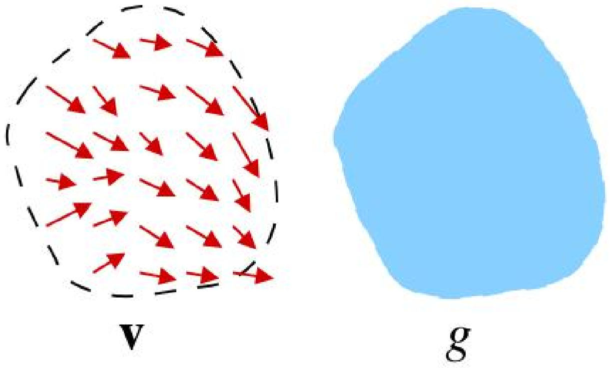
*f**, a known scalar function defined over *S* minus the interior of *Ω*

f, an unknown scalar function defined over the interior of *Ω*.

v, a vector field defined over *Ω*.

Guided interpolation notations





Unknown function f interpolates in domain Ω the target function $f*$, under guidance of vector field v, which might be or not the gradient field of a source function g.

Membrane interpolant

- The simplest interpolant *f* of *f** over *Ω* is the membrane interpolant defined as the solution of the minimization problem:

$$\min_f \iint_\Omega |\nabla f|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}, \qquad (1)$$

where $\nabla. = [\frac{\partial.}{\partial x}, \frac{\partial.}{\partial y}]$ is the gradient operator.

Euler-Lagrange Equation

- Using a fundamental equation of calculus of variations, Euler-Lagrange Equation, to solve the minimizer. Below is the result.

$$\Delta f = 0 \text{ over } \Omega \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}, \qquad (2)$$

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

Guidance field

- The route proposed in the Poisson image editing is to modify the problem by introducing further constraints in the form of a guidance field as explained below.
- A guidance field is a vector field *v* used in an extended version of the minimization problem (1) above:

$$\min_f \iint_\Omega |\nabla f - \mathbf{v}|^2 \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}, \qquad (3)$$

Poisson Equation

- The solution of (3) is the unique solution of the following Poisson equation with Dirichlet boundary conditions:

$$\Delta f = \operatorname{div}\mathbf{v} \text{ over } \Omega, \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}, \tag{4}$$

where $\operatorname{div}\mathbf{v} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}$ is the divergence of $\mathbf{v} = (u, v)$

- This is the fundamental machinery of Poisson editing of color images: three Poisson equations of the form (4) are solved independently in the three color channels of the chosen color space.

Discrete Poisson solver

- The variational problem (3), and the associated Poisson equation with Dirichlet boundary conditions (4), can be discretized and solved as below:

$$\text{for all } p \in \Omega, \quad |N_p| f_p - \sum_{q \in N_p \cap \Omega} f_q = \sum_{q \in N_p \cap \partial\Omega} f_q^* + \sum_{q \in N_p} v_{pq}. \quad (7)$$

- For each pixel p in S, let N_p be the set of its 4-connected neighbors which are in S, and let $\langle p,q \rangle$ denote a pixel pair such that $q \in N_p$.

Discrete Poisson solver

- Note that for pixels p interior to Ω, that is, $N_p \subset \Omega$, there are no boundary terms in the right hand side of (7), which reads:

$$|N_p| f_p - \sum_{q \in N_p} f_q = \sum_{q \in N_p} v_{pq}. \tag{8}$$

Importing gradients

- The basic choice for the guidance field *v* is a gradient field taken directly from a source image. Denoting by *g* this source image, the interpolation is performed under the guidance of

$$\mathbf{v} = \nabla g, \tag{9}$$

- and (4) now reads

$$\Delta f = \Delta g \text{ over } \Omega, \text{ with } f|_{\partial\Omega} = f^*|_{\partial\Omega}. \tag{10}$$

Importing gradients

- As for the numerical implementation, the continuous specification (9) translates into

$$\text{for all } \langle p, q \rangle,\ v_{pq} = g_p - g_q, \tag{11}$$

- which is to be plugged into (7).

Mixing gradients

- With the tool described in the previous section, no trace of the target image *f** is kept inside *Ω*. However, there are situations where it is desirable to combine properties of *f** with those of *g*, for example to add objects with holes, or partially transparent ones, on top of a textured or cluttered background.

$$\text{for all } \mathbf{x} \in \Omega,\ \mathbf{v}(\mathbf{x}) = \begin{cases} \nabla f^*(\mathbf{x}) & \text{if } |\nabla f^*(\mathbf{x})| > |\nabla g(\mathbf{x})|, \\ \nabla g(\mathbf{x}) & \text{otherwise.} \end{cases} \quad (12)$$

Mixing gradients

- The discrete counterpart of this guidance field is:

$$v_{pq} = \begin{cases} f_p^* - f_q^* & \text{if } |f_p^* - f_q^*| > |g_p - g_q|, \\ g_p - g_q & \text{otherwise,} \end{cases} \tag{13}$$

Implementation(Opaque)

- Let p_1 , p_2 , … p_i … p_{n-1} , p_n be *n* unknown pixels in target image.
- For each *p* interior to *Ω* , there is a equation

$$4f_p - f_{q1} - f_{q2} - f_{q3} - f_{q4} = 4g_p - g_{q_1} - g_{q_2} - g_{q_3} - g_{q_4}$$

- If *p* is not interior to *Ω* , e.g. , *q1* is out of *Ω*

$$4f_p - f_{q2} - f_{q3} - f_{q4} = 4g_p - g_{q_1} - g_{q_2} - g_{q_3} - g_{q_4} + f^*_{q1}$$

Direct

Opaque

Transparent

Direct

Opaque

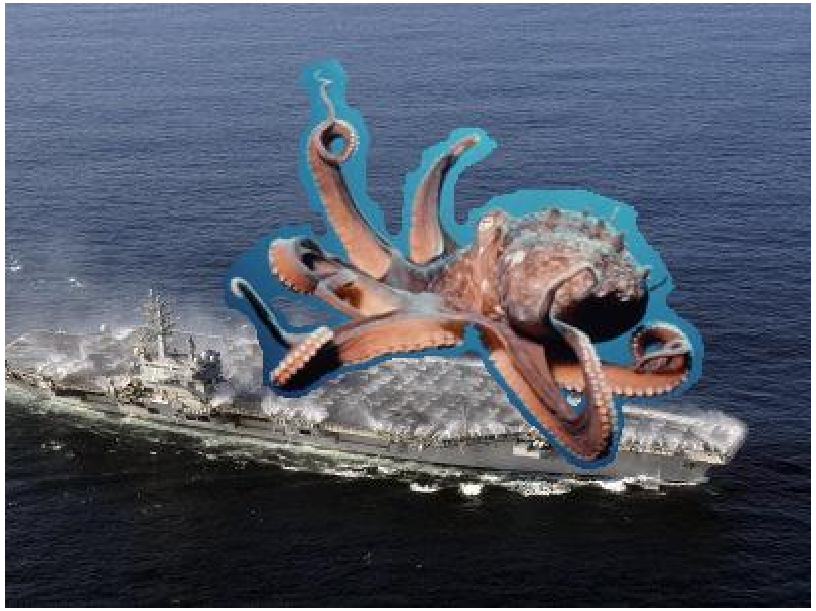
Transparent

Direct 

Opaque 

Transparent 

Direct



Opaque

Transparent

Source code on my Github