Photo Clip Art

Jean-François Lalonde, Derek Home, Alexei A. Efros, Carsten Rother, John Winn, Antonio Criminisi
Carnegie Mellon University   Microsoft Research Cambridge

SIGGRAPH 2007

Manuel Rodriguez Ladron de Guevara
PhD Computational Design Student

Instructor: Jun-Yan Zhu, TAs: Yufei Ye, Viraj Mehta
16-726 Learning-based Image Synthesis

Carnegie Mellon University

Problem

Image manipulation and edition requires skills, time and patience

Proposal

System for inserting new objects into existing photographs in an automatic fashion, as a:

- Data driven
- 3D based
- Context-sensitive

Object Retrieval Task

Some tasks involved in image composition:
- Search for good object
- Cut out
- Manual cropping
- Resizing
- Color adjustment
- Blending,
- Etc.

User is asked to do 2 things

Pick a 3D location in the scene to place a new object

Select an object to insert using a hierarchical menu

The approach is based on two critical elements

Data-driven Object Placement

3D Scene-based Representation

Approach

Data-driven Object Placement

The idea is to generalize the object insertion.

Instead of a particular object (e.g. my brown Volvo seen from the side), use general instance of an object class (e.g. car)

Instead of manipulating object (color, geometry, orientation), retrieve an object of a specified class that has all required properties :

- Camera pose
- Lighting
- Resolution
- Etc.

Approach

3D Scene-based Representation

Image manipulation in the 3D space of the scene, not in 2D space of the image.

3D space of scene includes:

- Rough qualitative information about surfaces
- Surface orientations w.r.t camera
- Basic depth ordering

Library: objects are annotated with:

- Relative camera pose
- Occlusion (whether they are occluded or not by other objects)
- Ground plane
- Etc.

Key challenges

Rich Object Library: Large number of labeled objects. LabelMe dataset (18 categories 13,000 object instances after processing data)

Object Segmentation: Automatic segmentation and blending approach.

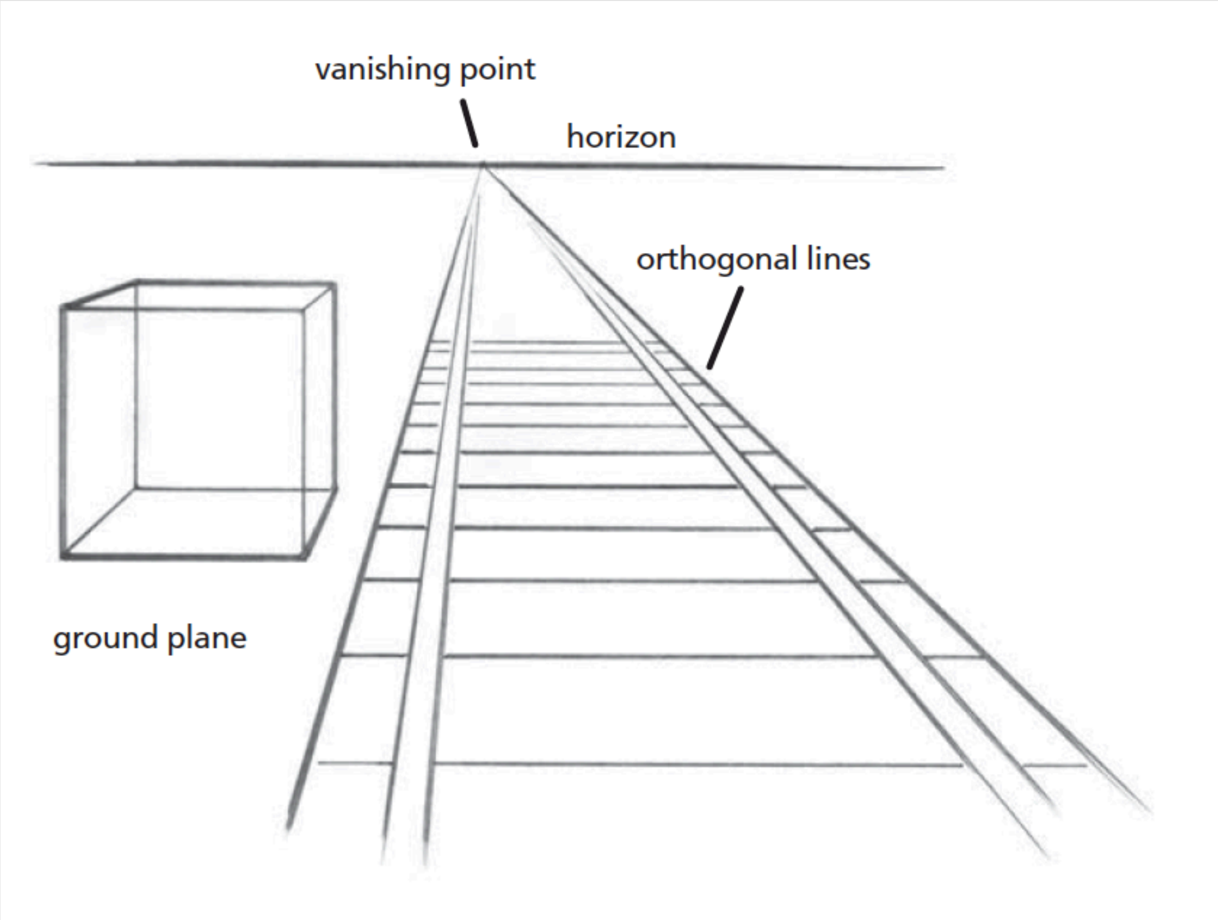
True Object Size and Orientation: Automatic algorithm to estimate camera height and pose w.r.t ground plane.

Lighting Conditions: Illumination model based on color distributions of major surfaces

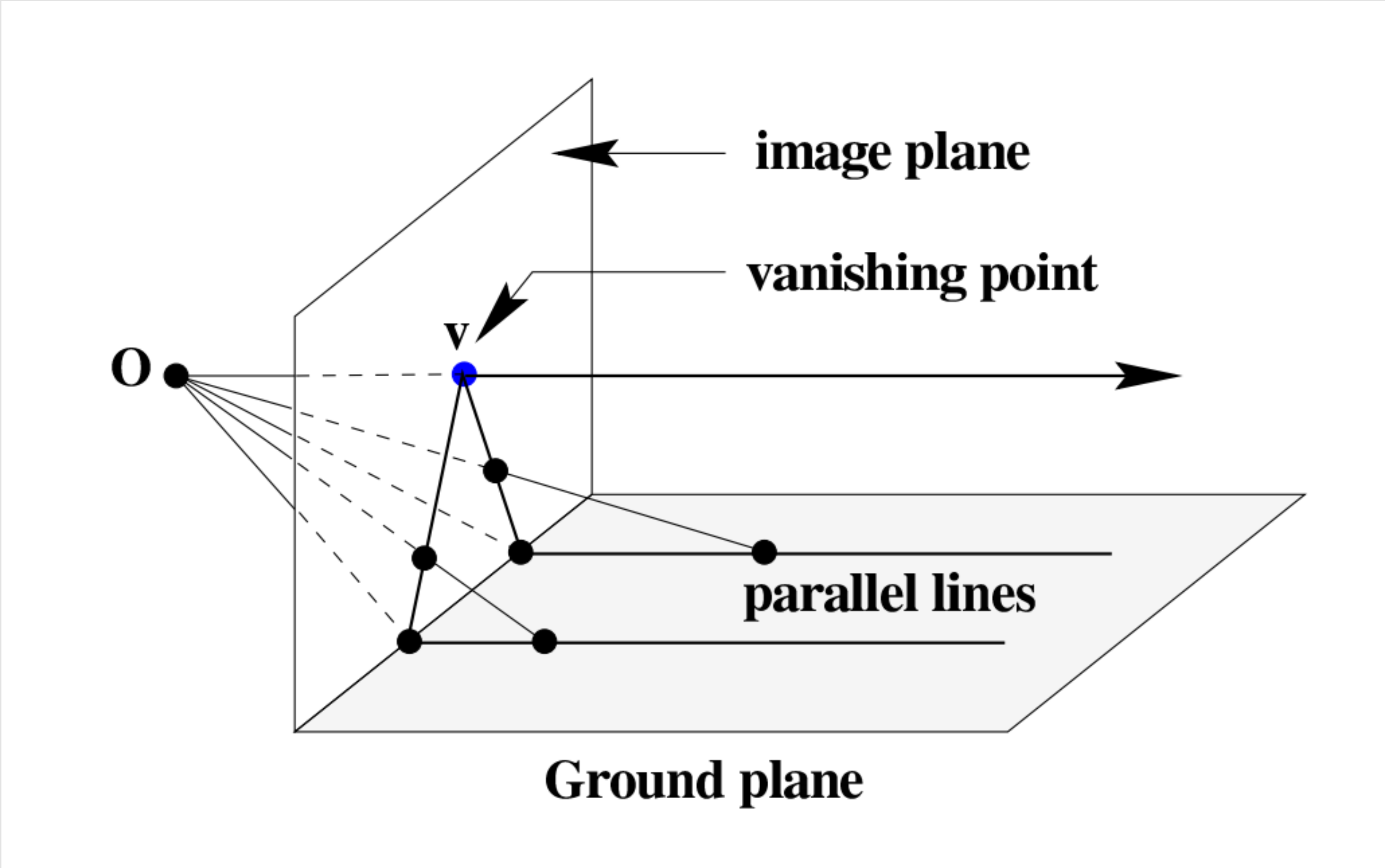
Intuitive User Interface

Estimating object size and orientation

- Vanishing line of the ground plane gives enough information to estimate relative heights of **objects that rest on the ground plane**



https://www.artistsnetwork.com/art-mediums/drawing/learn-to-draw-perspective/



https://computergraphics.stackexchange.com/questions/5793/why-is-the-line-from-the-camera-to-vanishing-point-parallel-to-the-other-paralle

Estimating object size and orientation

- Vanishing line of the ground plane gives enough information to estimate relative heights of **objects that rest on the ground plane**

$$y_i = \frac{h_i y_c}{v_o - v_i}$$

* Assuming that objects stand on the ground and the ground is not tilted from side to side and roughly orthogonal to the image plane

y_i = 3D height of an object
h_i = 2D height of the same object

y_c = camera height
v_0 = horizon position
v_i = vertical position (measured from the bottom of the image)

Estimation

Position and height of 2 objects of known type in the image is enough to:
- estimate the camera pose without relying on a prior distribution
- compute the 3D height of other objects in the scene

Estimating object size and orientation

- Vanishing point of the ground plane gives enough information to estimate relative heights of **objects that rest on the ground plane**

$$y_i = \frac{h_i y_c}{v_o - v_i}$$

* Assuming that objects stand on the ground and the ground is not tilted from side to side and roughly orthogonal to the image plane

y_i = 3D height of an object
h_i = 2D height of the same object

y_c = camera height
v_0 = horizon position
v_i = vertical position (measured from the bottom of the image)

Algorithm

1) Compute camera pose for images that contain at least 2 known objects (with a known height distribution)
2) Estimate prior distribution over camera pose
3) Infer camera pose for images that contain only 1 known object.
4) Compute heights of objects in the images for which camera pose is known and estimate the height distributions of new classes of objects.
5) Repeat for more objects

Estimating object size and orientation

Example

Initialize height of people with mean 1.7 and standard deviation of 0.085

Initialize loose prior of camera pose with mean horizon of mid-image and camera height of 1.6 (eye level), with diagonal covariance of 1.

Once camera pose is estimated for images containing at least 2 people, use those poses to model the prior with a Gaussian mixture model.

Then infer the height of at least 15 instances of an object class and estimate the normal distribution of heights for that object.

Figure 2: Automatic object height estimation. Objects taken from a typical image in LabelMe dataset (left) are first shown in their original pixel size (center), and after being resized according to their automatically estimated 3D heights (right).

Estimating lighting conditions

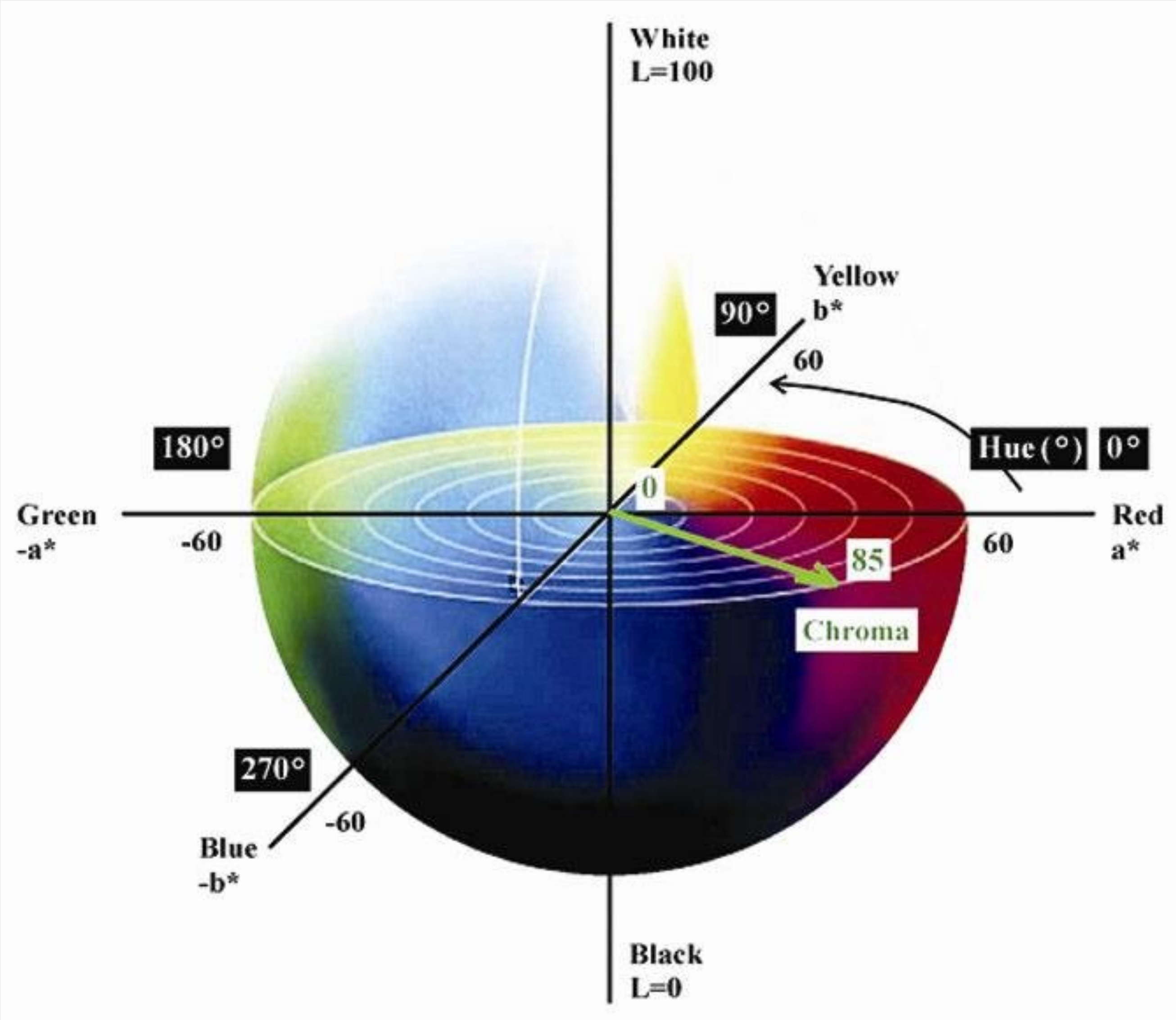
Idea

Rough 3D structure of the depicted scene and use it to collect lighting information from the 3 major zenith angle directions:

- From above
- From below
- From the sides

Automatic estimation of 3 major surface types: ground plane, vertical planes and sky.

Distribution of illuminations within each surface type is computed as a joint 3D histogram of pixels colors in the CIE L*a*b* space, for a total of 3 histograms to form the **scene illumination context.**



DETERMINATION OF THE EFFECTS OF DIFFERENT PACKAGING METHODS AND MATERIALS ON STORAGE TIME OF DRIED APPLE. Recep Kulcu, MATTER:International Journal of Science and Technology

Estimating lighting conditions

Figure 3: Lighting plays crucial role in object insertion. Given an input image (a), objects that are similarly illuminated fit seamlessly into the scene (b), while these with substantially different illumination appear out of place (c). By considering the source image from which an object was taken (d) in relation to the input image (a), our algorithm can predict which objects will likely make a realistic composite.

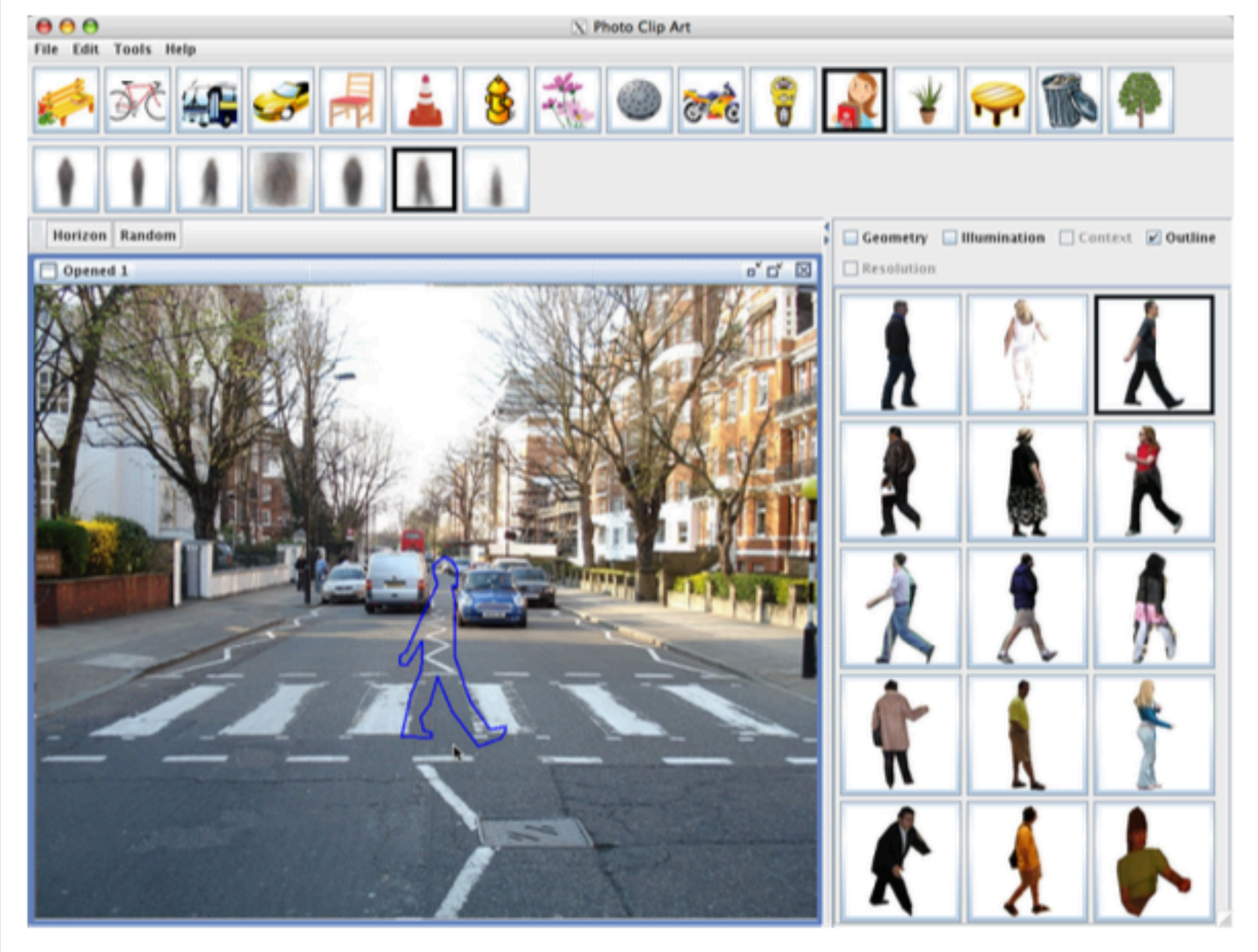
Filtering and Grouping

Remove non-whole objects

Estimate 3D height of all objects using the iterative procedure described previously

Clustering is used to automatically find visually similar subclasses for each object class:

- Objects are rescaled to their 3D height (so tree and bush are not clustered together)
- K-means clustering using L2 norm on binary object outlines



Object Insertion

User Interface

Matching Criteria

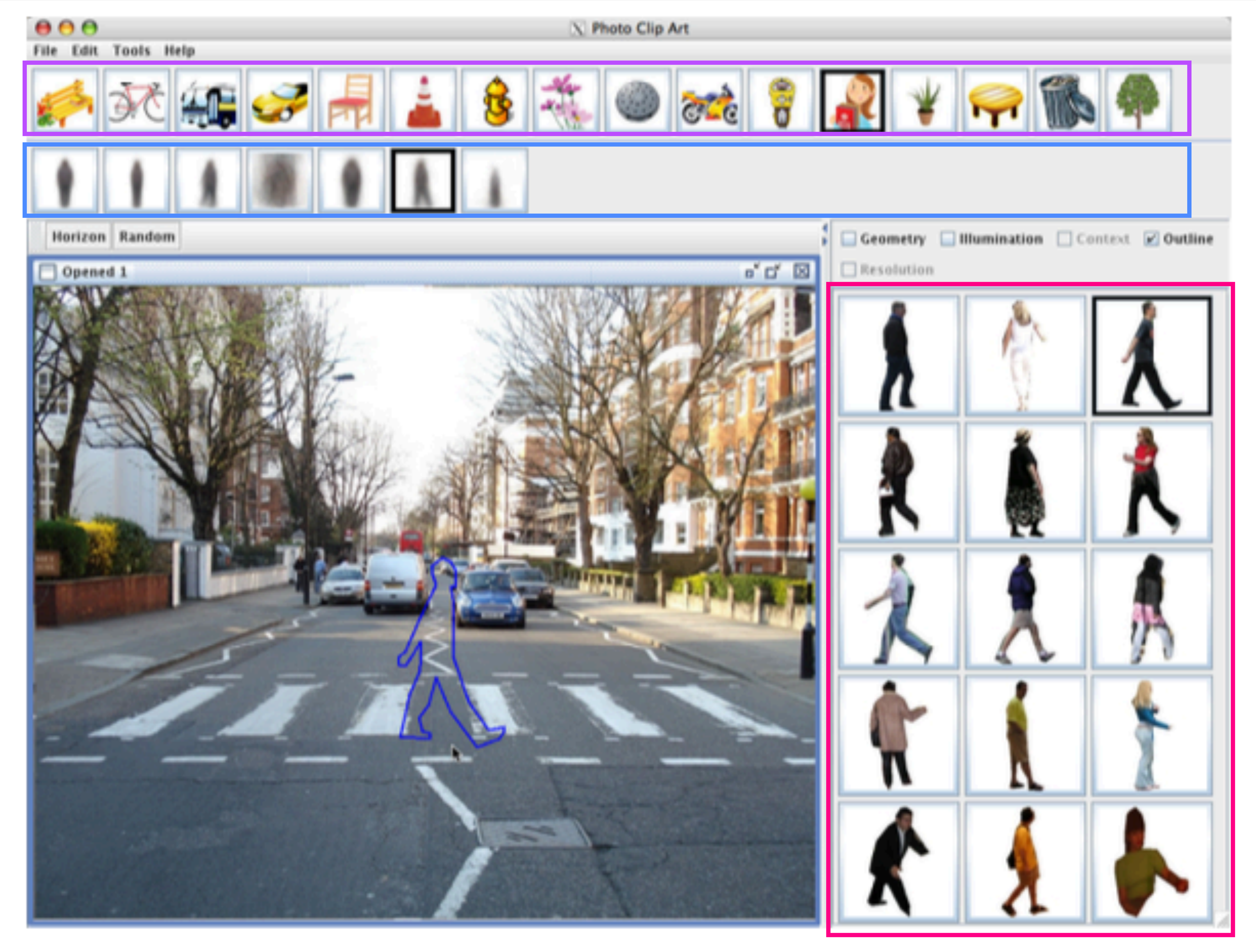
Object Segmentation and Blending

Shadow Transfer

User Interface

Given new photograph:

1) Pick horizon line
2) Navigate
3) Paste object of choice with single click



Object Class

Object Subclass

Object Instances, sorted by how well they match, top to bottom

User Interface

Given new photograph:

1) Pick horizon line
2) Navigate
3) Paste object of choice with single click

Matching Criteria

(Sorted instances)

Camera Orientation (pre-computed)

Global Lighting Conditions: an object is most likely to appear photometrically consistent with he new scene if its original scene illumination context match that of the new scene.

Distance between illumination contexts is computed as a weighted linear combination of the squared distances between the L*a*b* histograms for sky, vertical and ground

Local Context: if user specifies the location first, local context is computed using SSD.

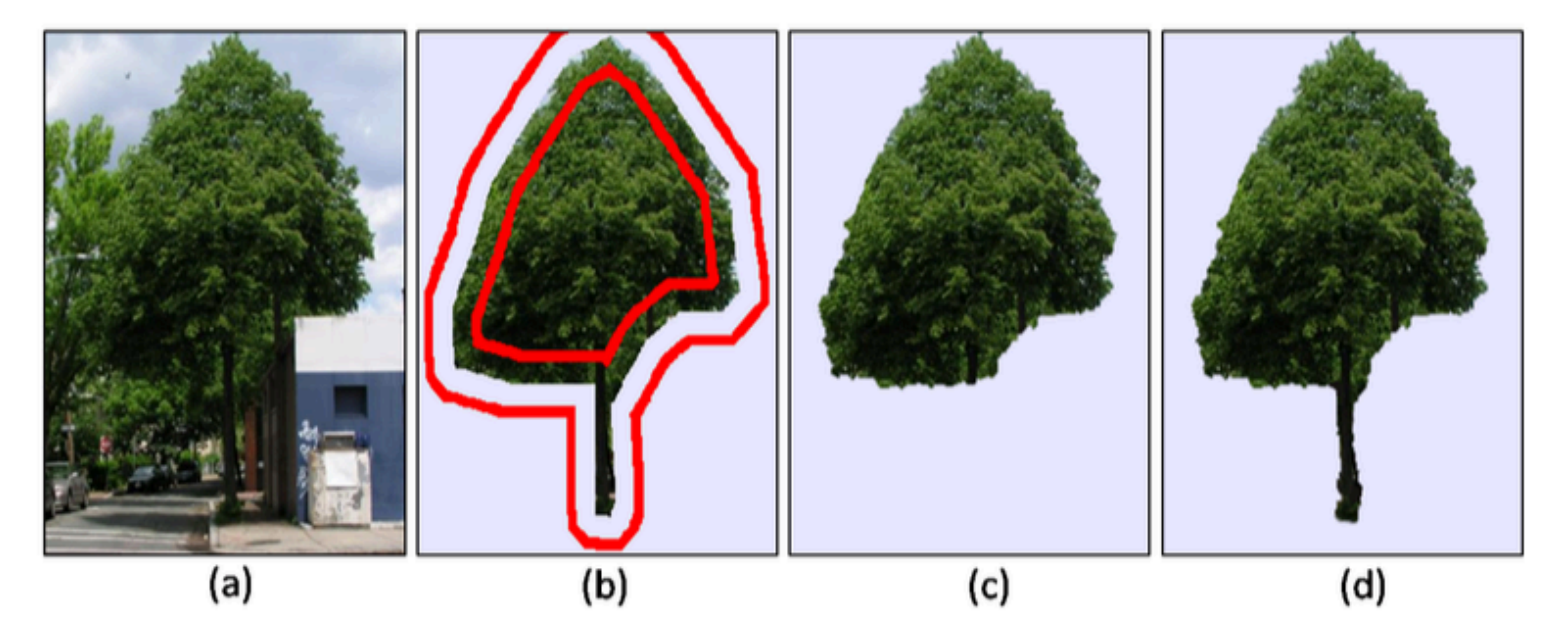
Resolution

Segmentation Quality

Object Segmentation and Blending

Shape Prior: shape prior that overcomes *shrinking bias* without over-smoothing the whole segmentation.

Idea: gradient of the contour is similar to gradient of given shape.



Input image

Crude polygon-shaped LabelMe segmentation

Shrinking bias (by algorithms like GrabCut)

Results introducing shape prior

*Algorithm and equation details in paper

Object Segmentation and Blending

Context-sensitive Blending

Idea: find binary blending mask which either blends in the background color or leaves the segmentation unaltered

(a) Input image | (b) GrabCut [Rother et al '04] | (c) Poisson Editing [Perez et al '03] | (d) Blending Mask - Our Method | (e) Our Method | (f) Digital Photomontage [Agarwala et al '04] | (g) Drag-and-Drop Pasting [Jia et al '06]

Two artifacts:
1) severe discoloration
2) blurry halo

Where mask coincides with object (red line), no blending is performed. When the mask is outside the object, it blends in the background

How?
1) Add additional regional term that measure the similarity of foreground and background statistics (green line). This allows the white mask to grow outside the red mask at places with high similarity
2) Modify standard Poisson image blending to prevent discoloration using the following formula:

$$E(u) = \lambda \boxed{\int_u (u - I^F)^2} + \int_s w_s(\nabla I^F)||\nabla u - \nabla I^F||^2$$

Forces the object to retain its original color

I^F Image foreground

u Constructed image

Where $w_s(\nabla I^F) = 1 + \gamma exp - \frac{\beta}{2g_s}||\nabla I^F||$, with $g_s = mean(||\nabla I^F||^2)$

$\lambda = 0.05, \gamma = 800, \beta = 0.5$

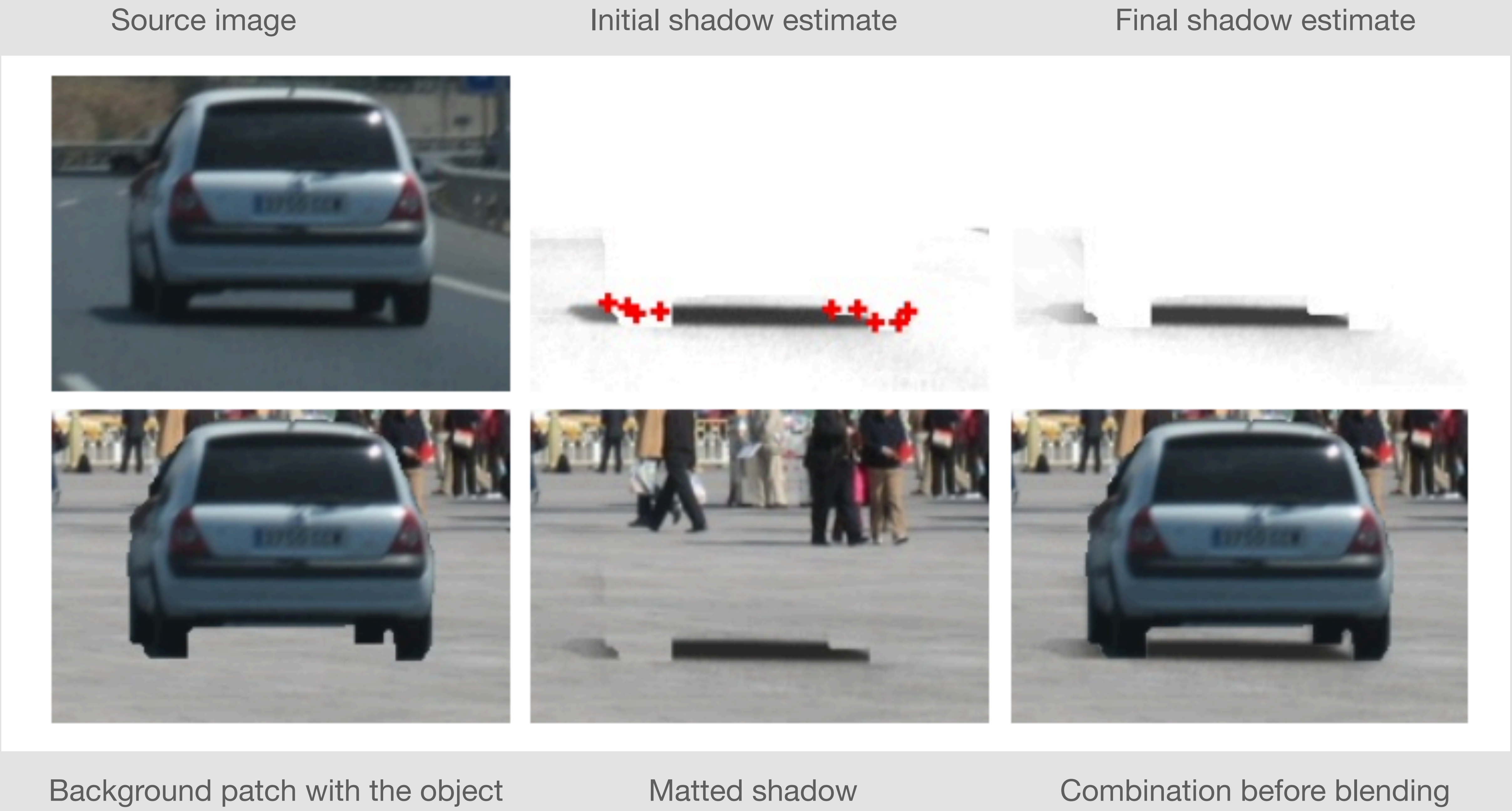
Object Segmentation and Blending

Transferring Shadows

Idea: image-based approach. Determine percentage intensity drop at each image position due to the object's shadow.

Initial shadow estimate: pixel intensity divided by the non-shadowed ground intensity.

Non-shadowed ground intensity is approximated on each row as the median intensity of pixels within a margin around the object, excluding pixels within and directly beneath the object region.



Results

Results

Failures

Thank you

Manuel Rodriguez Ladron de Guevara
Instructor: Jun-Yan Zhu, TAs: Yufei Ye, Viraj Mehta
16-726 Learning-based Image Synthesis