Part Annotations via Pairwise Correspondence

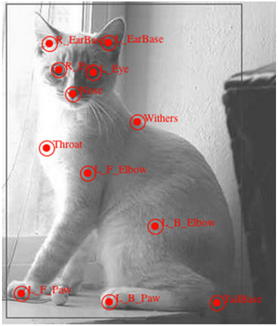
Subhransu Maji, Greg Shakhnarovich

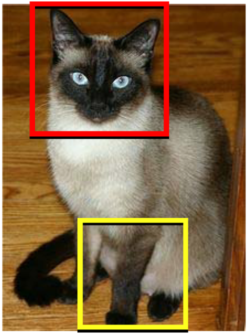
TTI–Chicago

July 2012

Part annotation

- Many modern vision methods rely on “parts”
- Typical annotation paradigm: mark a set of parts, identified by name and description





Diverse categories

- Appearance variability
- Structural flexibility
- Unnameable or unnamed landmarks
- But: we know two matching parts when we see them

Annotation task on AMT

Mark common landmarks between the two churches

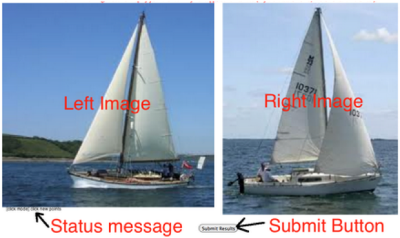
See this page for how to use this interface. See examples here.

NOTE: If the image does not display your browser may not support this interface (Try the latest Chrome, Firefox or Safari browser).

Submit Results

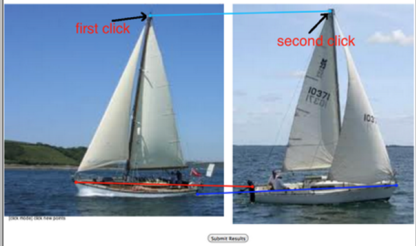
Layout and instructions

Interface Layout



The interface contains two images (left and right). In the bottom left is the status message and in the bottom center is the submit button. Your task is to click points on the left image and their corresponding points on the rigth image.

How to add points?

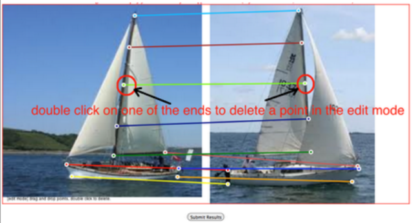


To add a point first click on the left image. This starts the line and then click again on the right image. *You have to click on the left image first*. The image above shows three pairs of points added. *Note: to edit or delete points see below*.

Edit mode

To edit points such as delete and move them, you have to be in the edit mode. Press "e" to toggle between edit mode and click mode. In the edit mode you see that the images are surrounded by a red border. The end points of the lines are marked with white circles and the status message also says "edit mode". To add more points you have to go back to the "click" mode by pressing "e". You can do this any number of times.

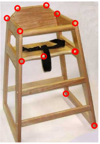
How to delete points?



To delete a pair of points double click on one of the end points in the edit mode. The pair gets removed from the interface. *Note that you have to be in the edit mode.*

Examples to workers

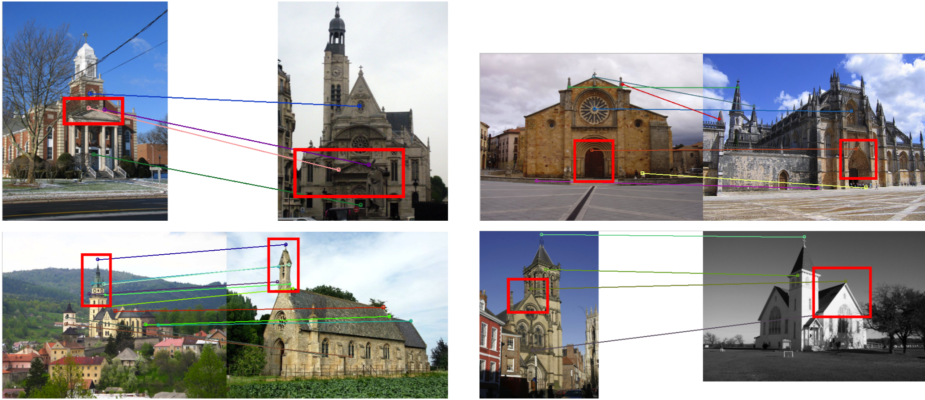
- Provide a handful of examples of landmarks



- Make it clear this is non-exhaustive!

Example annotations and correspondences

- 1000 pairs over approx 300 images
- Median: 3 annotations, 48 seconds per pair

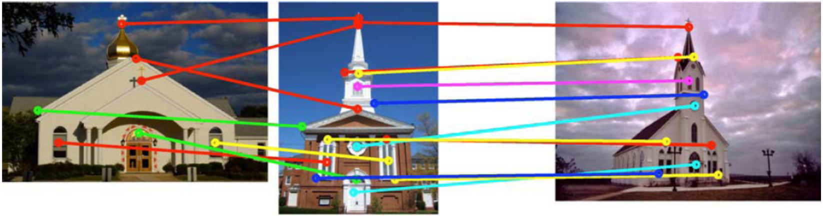


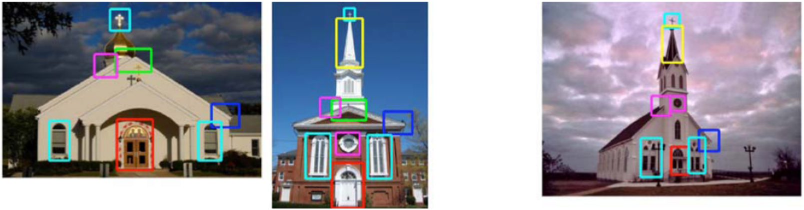
Example annotations and correspondences

- 400 pairs over 250 images, 3-5 workers per pair
- Median: 2 annotations, 34 seconds per pair

Propagating correspondences: semantic graph

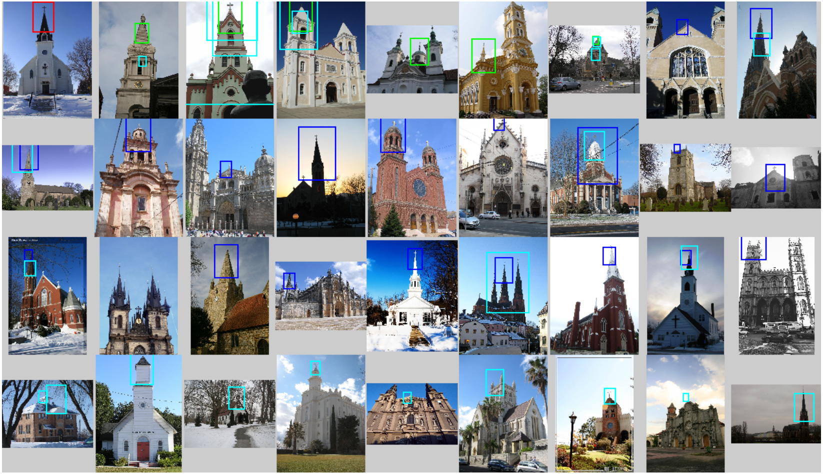
- Image $\Leftrightarrow$ vertex, edge $\Leftrightarrow$ pairwise annotation
- Can traverse the graph to infer new correspondences





Inferred correspondences

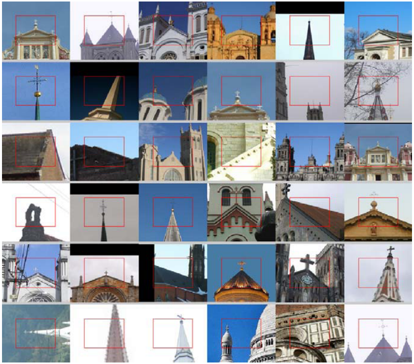
Propagating pairwise correspondences



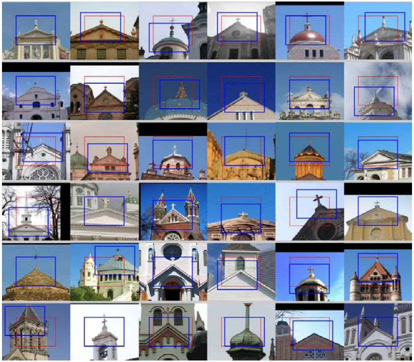
- Source (red), depth 1 (green), 2 (blue), 3 (cyan)
- Noisy; can't apply to new images – need to learn appearance

Learning parts with latent SVM

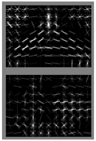
graph only



graph and appearance



appearance HOG fiter

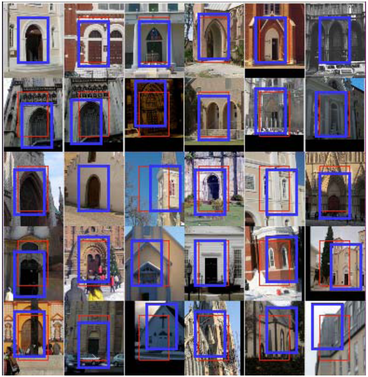


red: initial, blue: final (learned)

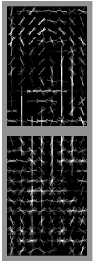
Another visual part

graph only

graph and appearance



filter



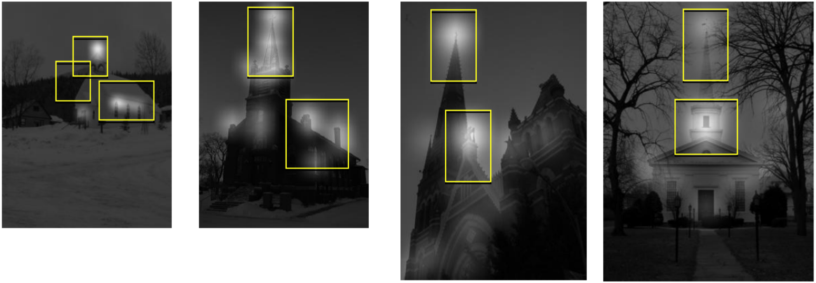
red: initial, blue: final

Semantic saliency

- Annotation density provides a measure of saliency:

Saliency-guided exploration

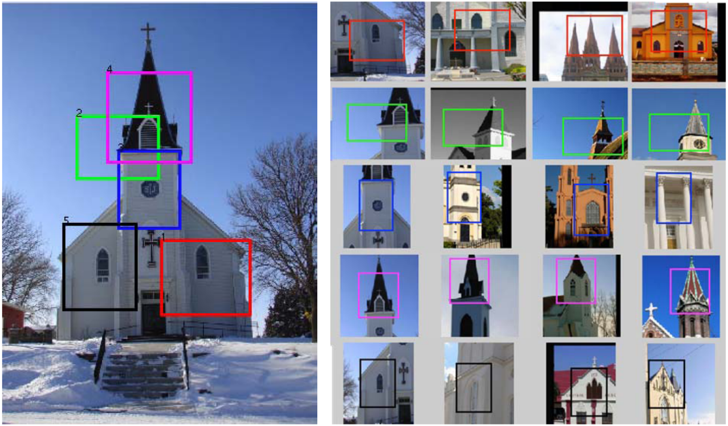
- We can sample windows (part candidates) according to saliency
- Use them to learn part appearance model
- Select a subset based on desired criteria: diversity, accuracy, parsimony. . .



Library of parts

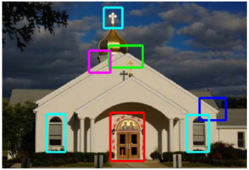
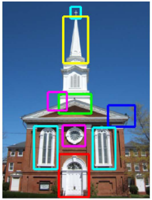
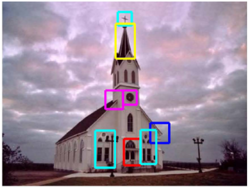
What can we do with parts

- Building blocks for rich part-based representation
- Parsing an instance in terms that relate it to others in the category:



Conclusion: Pairwise correspondence annotation

- Extremely easy to set up and deploy
- Less affected by designer's bias
- Exploits rich semantic knowledge of annotators
- More robust to inconsistencies
- Applicable to structurally and visually diverse categories
- Starting point to learning rich representations in computer vision

intentionally left empty

Latent SVM appearance model

- Optimal subwindow L in a training image is a hidden variable
- Response of the model for part p in image I

$$F(I; \mathbf{w}_p) = \max_L \langle \mathbf{w}_p, \phi(I, L) \rangle$$

  $\phi(I, L)$ is feature vector computer over subwindow L in image I
- Discriminative learning: given a negative (I_-, L_-) and a positive (I_+) for a part p, we strive for $F(I_+; \mathbf{w}_p) - F(I_-; \mathbf{w}_p) \geq 1$
- In our experiments: HOG features, L limited to windows overlapping initial location (inferred from the graph)