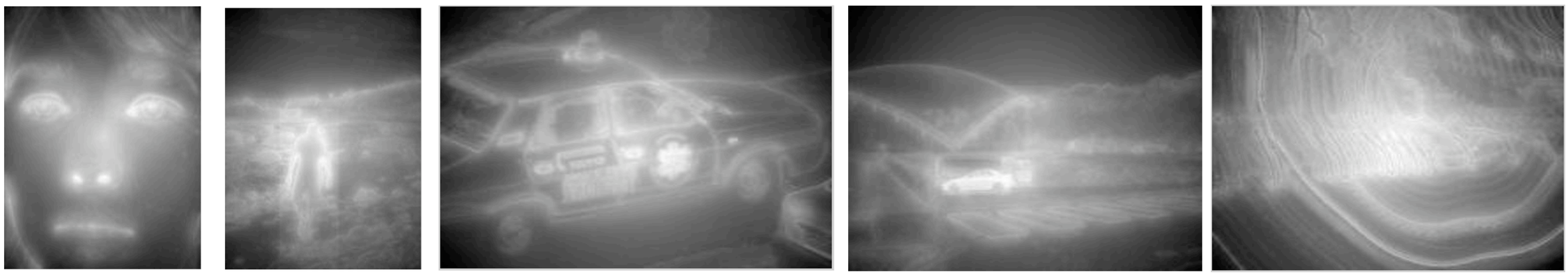
Learning to predict where people look

Tilke Judd, Antonio Torralba, Frédo Durand

Where do you look in these images?

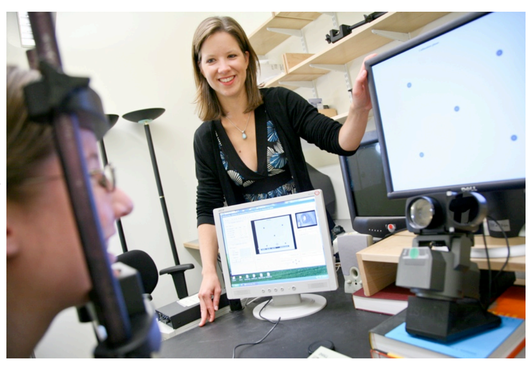
This is where other people looked in eye tracking tests.



This is where our model predicts you will look.

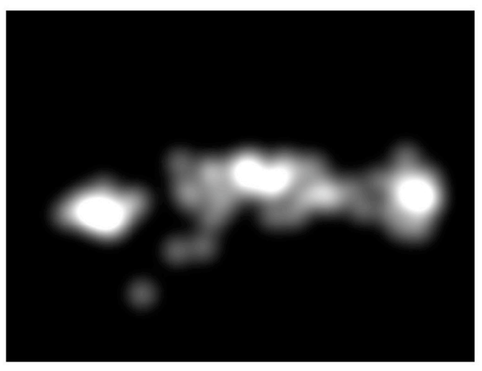
How do we do this?

Image database
We collected a large database of 1000 natural image from Flickr and LabelMe

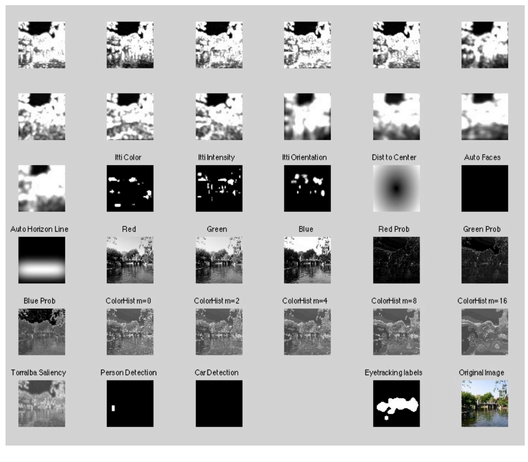


Eye tracking experiment and database
We ran a large eye tracking experiment with 15 users and 1000 images. This is the largest eye tracking database of natural images that we know about! and will be made available to the public.

Fixation information
Colored squares indicate locations that 15 viewers fixated on when viewing this photograph, We stored data about the path and timing of user's fixations through the image.

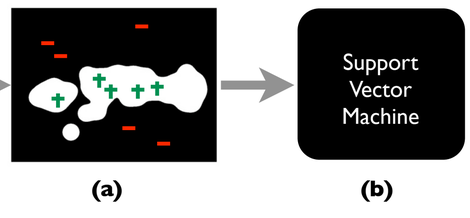


Human saliency map
We use the fixation locations from all 15 viewers to create a ground truth saliency map which shows the likelihood of a human to look at a certain location.



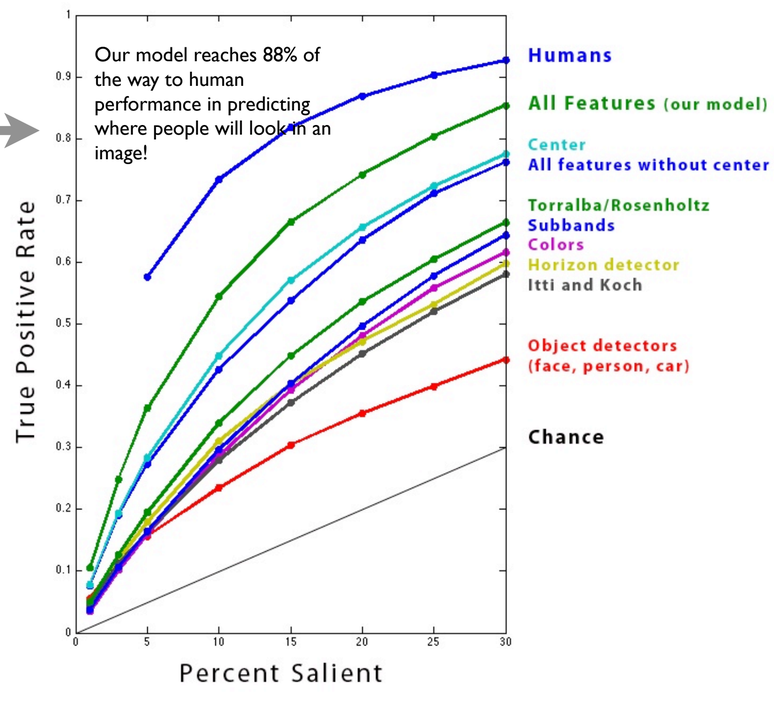
Features
We collect a set of features we believe might be predictive of where people look. These include:
low level image features
- illuminance, color, and orientation

high level image context features
- location of the horizon line,
- distance to the center of the image,
- presence of a face, person, or car.



(a) Training Samples
On a subset of images we chose several salient and non-salient locations as training samples. For each sample we have a label and a vector of feature values.

(b) Learning a Model
We use our training samples to train linear models using a support vector machine. The models aim to find weights for combining features that leads to the most accurate prediction of the saliency label. We test the models on the remaining images in our database to asses performance.



Performance Results
This ROC curve compares the performance of models trained on different sets of features. The y axis indicates the percentage of human fixations that lie inside the area of an image predicted as salient by a model.

Why do this?
For applications in graphics, smart design, human computer interaction:

- automatic image cropping or thumbnailing
- direct foveated image compression
- suggest levels of detail in non-photorealistic rendering