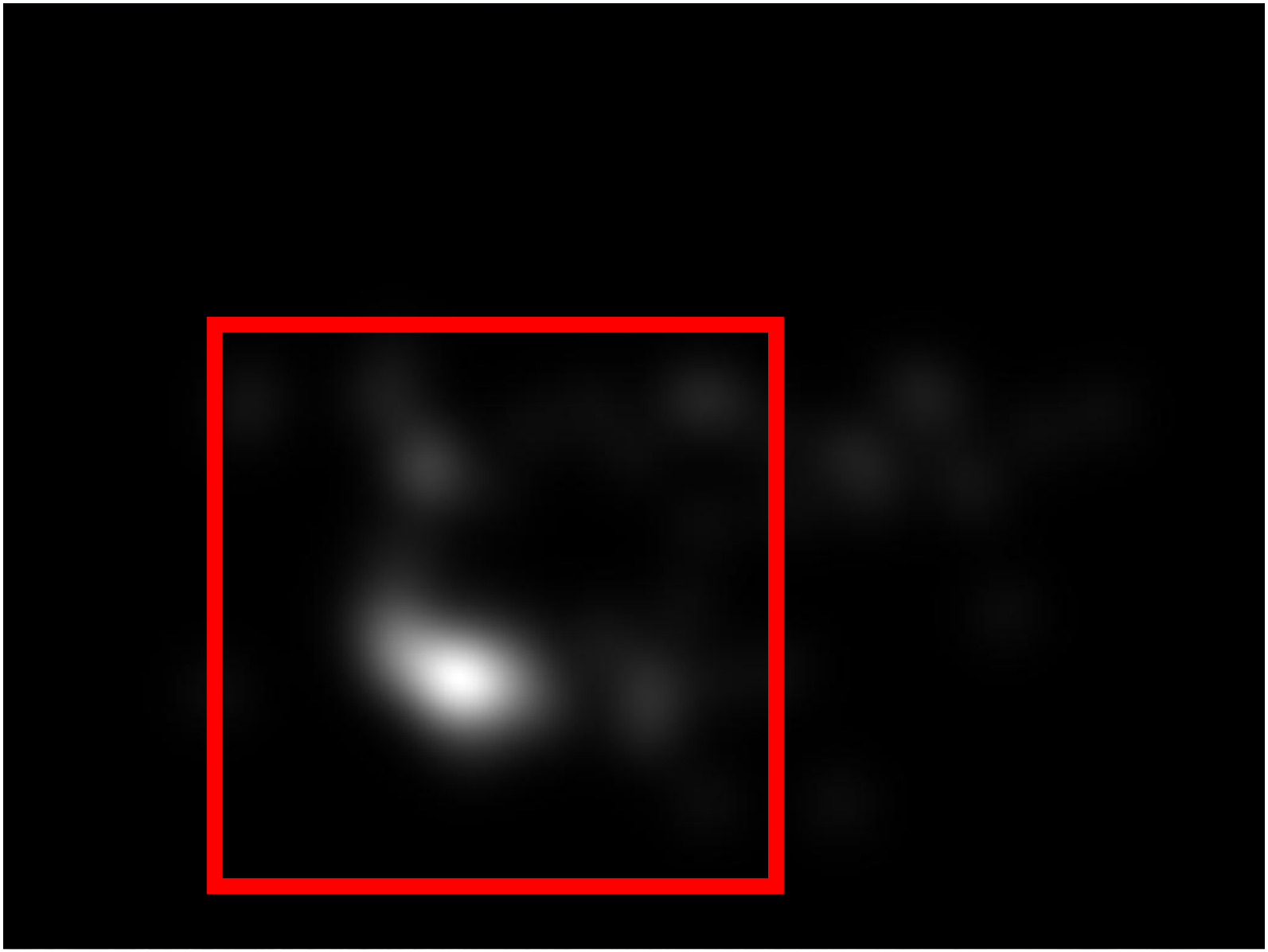
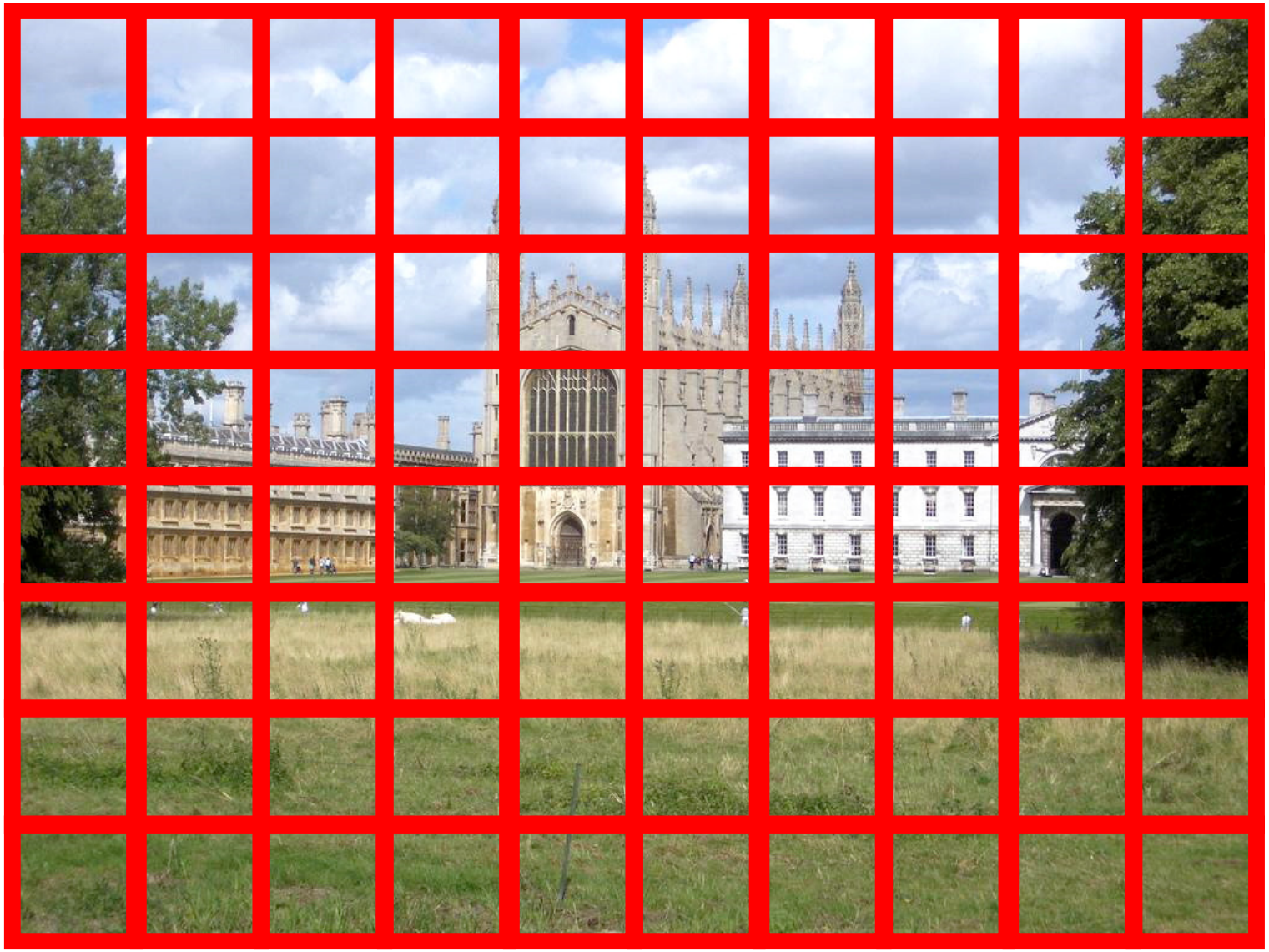
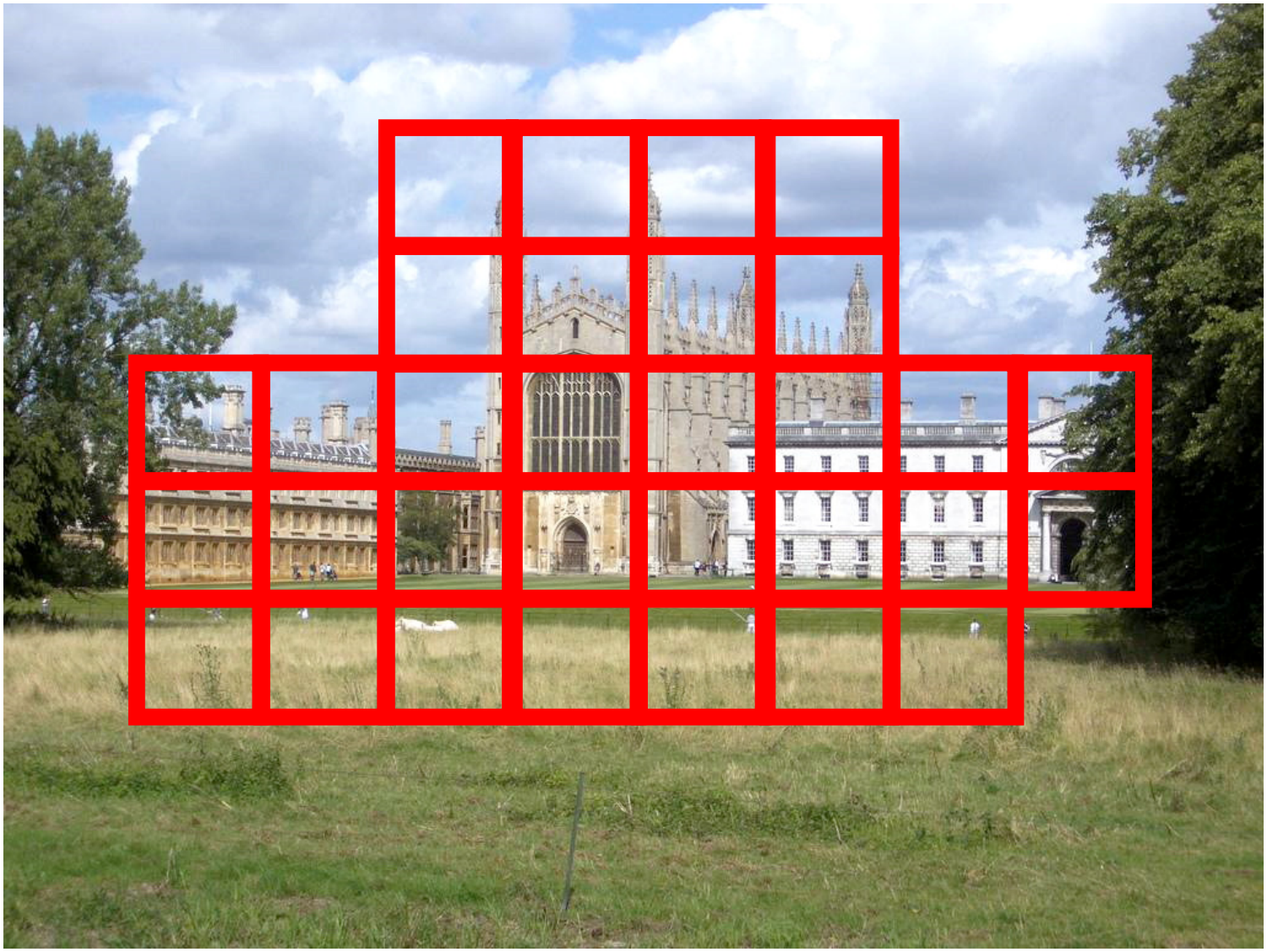
Learning to Predict Where Humans Look

Presented by: Chris Thomas
Paper By: Tilke Judd, et al.

Presentation Slides Loosely Based on Slides on site







What is common to these situations?

- In both situations:
  - Need to prioritize the visual field and decide what is important (i.e. salient)
- Why would you want to know where people look in an image?
  - Automatic Image Cropping (first example)
    - Extracting thumbnails on computer
  - Image search
  - Compression
  - Advertising design
  - Non-photorealistic rendering
  - Object detection
  - Self-driving car / robot navigation
  - …. Many other applications
- How do you get the data?
  - Eye-tracking data = Expensive / time consuming
  - Computational approaches – Not as accurate

heat map of 200 people

www.thinkeyetracking.com

6% of people look at the package

www.thinkeyetracking.com

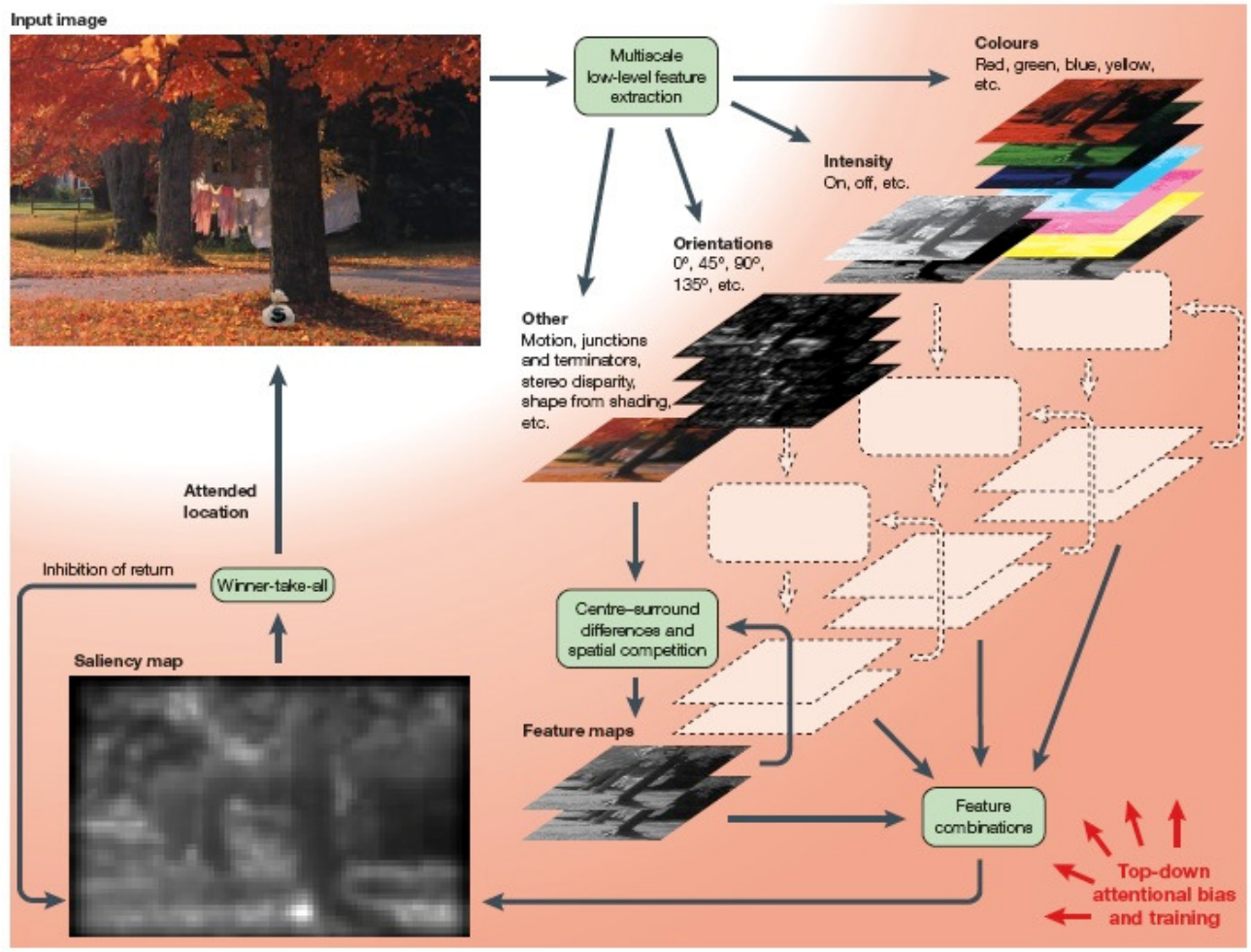
84% of people looked at the package

What is Saliency?

- Saliency is the quality that makes certain areas, objects, regions, etc. in the visual field stand out from the rest of the field and grab our attention
- This ability has developed evolutionarily
  - Critical to rapidly find predators, prey, etc.
  - Information overload in the visual field - too much to process at once
  - Solution: Do higher level processing serially
  - Problem: How do you know what to look at?
- Neuroscience research has shown that early visual neurons are specialized to detect certain types of basic "features" compared to their surroundings
  - Orientation contrast
  - Color contrasts
  - Lumination contrasts
  - Motion
  - Flicker

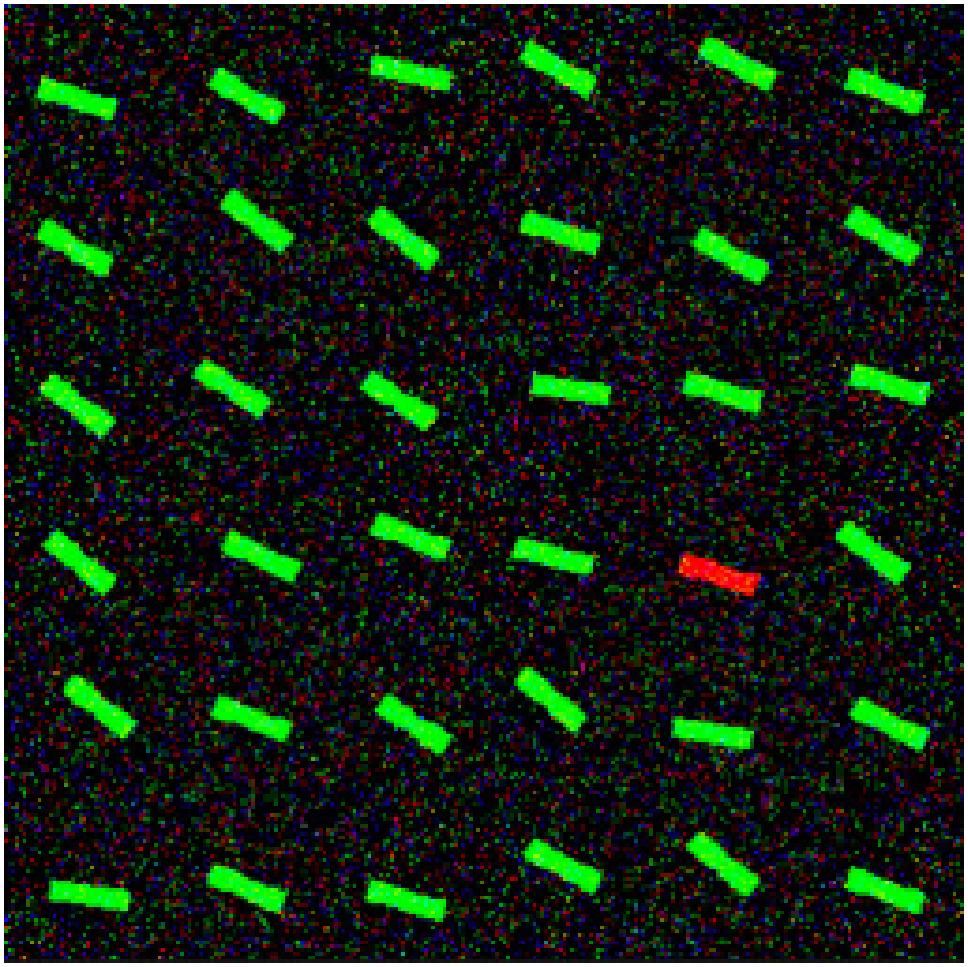
Bottom-Up Saliency Models

- Often based on biologically plausible linear filters
- Measure intensity, illumination, color, flicker, orientation contrast
- Lots of parameters need to be tuned
- Bottom-up
- Doesn't always predict fixations well

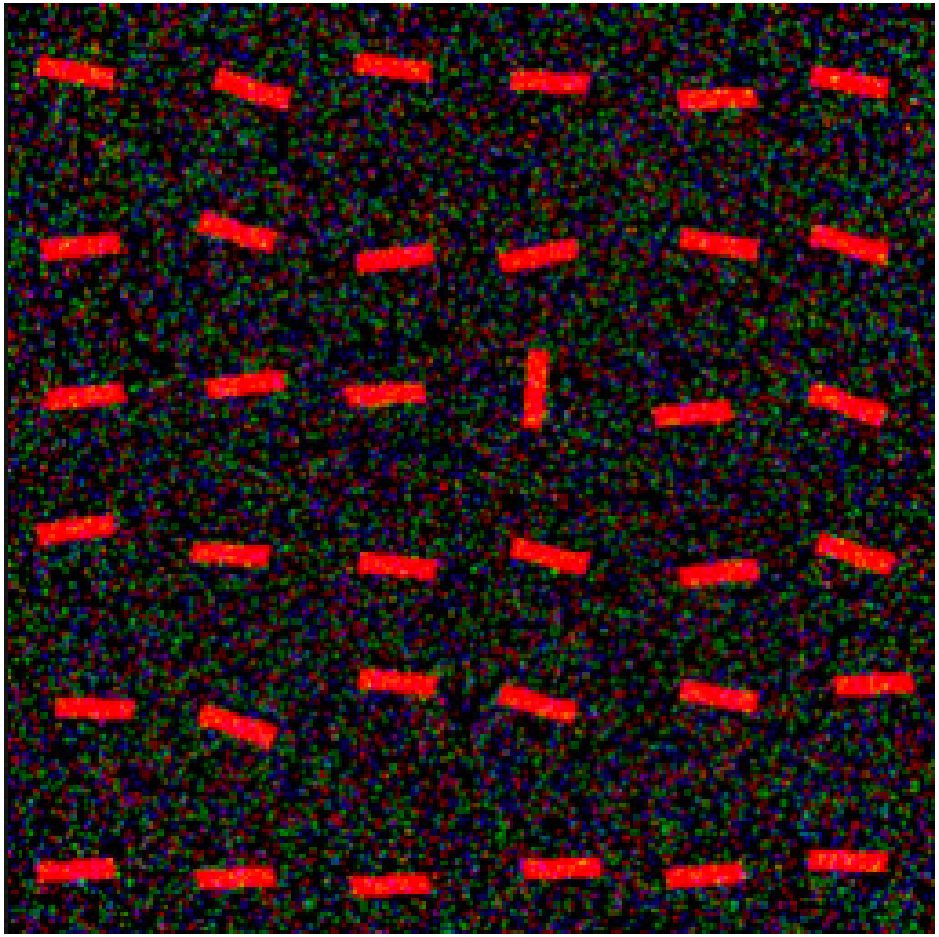


Bottom-up saliency model. From Itti and Koch [2001]

Bottom-Up Influences



Bottom Up Influences



Find the pedestrian


FIREHOUSE 1915 BREWING CO.
RAPID CITY FIRE DEPT
FIREHOUSE BREWING CO. RUSH ROOM

There is no pedestrian


FIREHOUSE
1915
BREWING CO.
RAPID CITY FIRE DEPT

Where do people actually look?

- Top-down task and scene dependent cues
  - Semantic understanding, memories, task
- Bottom-up saliency cues

Many models for saliency exist

Biologically Inspired

Itti and Koch (1998)

Cerf et al. (2007)

Hou and Zhang (2007)

Rosenholtz (1999)

Itti and Baldi (2006)

Le Meur et al. (2006)

Seo & Milanfar (2009)

Zhang & Cottrell (2008)
SUN model

Goferman et al. (2009)

Achanta (2010)

Mathematically Inspired

Heral et al. (2007)
Graphical Model

Avraham and Lindenbaum (2009)
Esaliency

Bruce and Tsotsos (2009) Information theoretic approach

Kienzle et al., (2007)

Gao and Vasconulos (2005)

Itti and Baldi (2006)
"Surprise" model

Navalpakkam and Itti (2005)

Elazary and Itti (2010)

Add top-down features

Ehinger et al., (2009)
(search task)

Oliva et al. (2003)

Torralba et al. (2006)

Zhang et al. (2008)

Kanan et al. (2009)

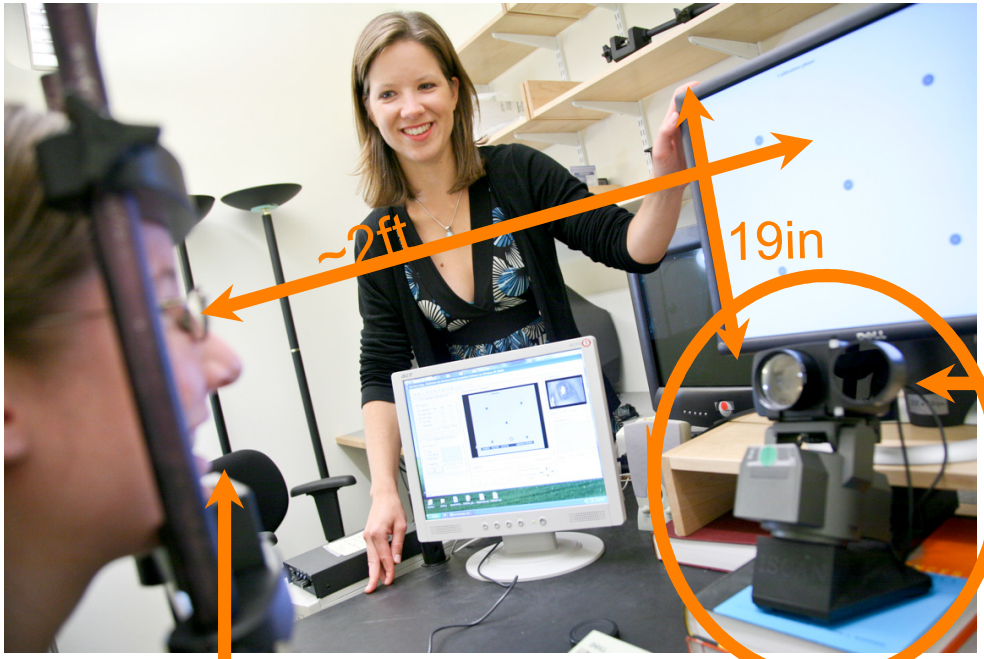
Problems and Contributions of Paper

- Many of the bottom up saliency models have many, many parameters that need artificially tuned
- Top-down attention is ignored is many models
- Lots of models exist, but no large standard benchmark existed
- Paper aims to Address these Problems:
  - Integrate many of the ideas from many existing bottom-up and top-down algorithms in one system trained using machine learning
  - Consider low-level, mid-level, and high-level visual features
  - Provide the largest scale eye-tracking dataset as a benchmark

- **Goal:** Learn where people look directly from the eye tracking data

1003 Natural Images from Flickr and LabelMe

Eye-Tracking Experiments



screen resolution
1280x1024

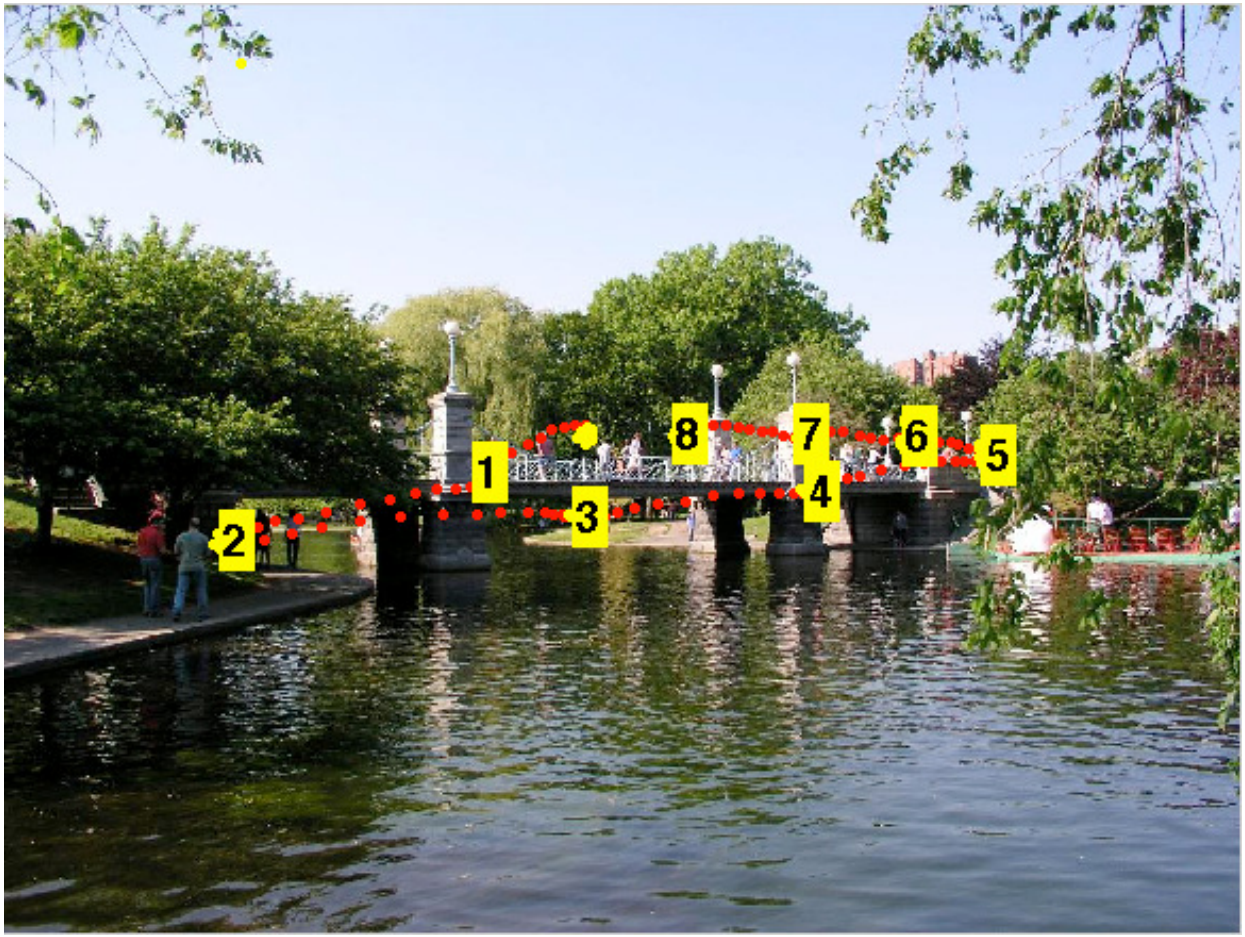
each image
shown for 3
seconds

eye tracker
measures
location of eye
fixation several
times a second.

user rests head in chin rest

- 15 People free viewed the images (actually done in dark room)
- Calibrate check every 50 images
- Divide the experiment into two sessions of 500 randomly ordered images
- Initial fixation is discarded

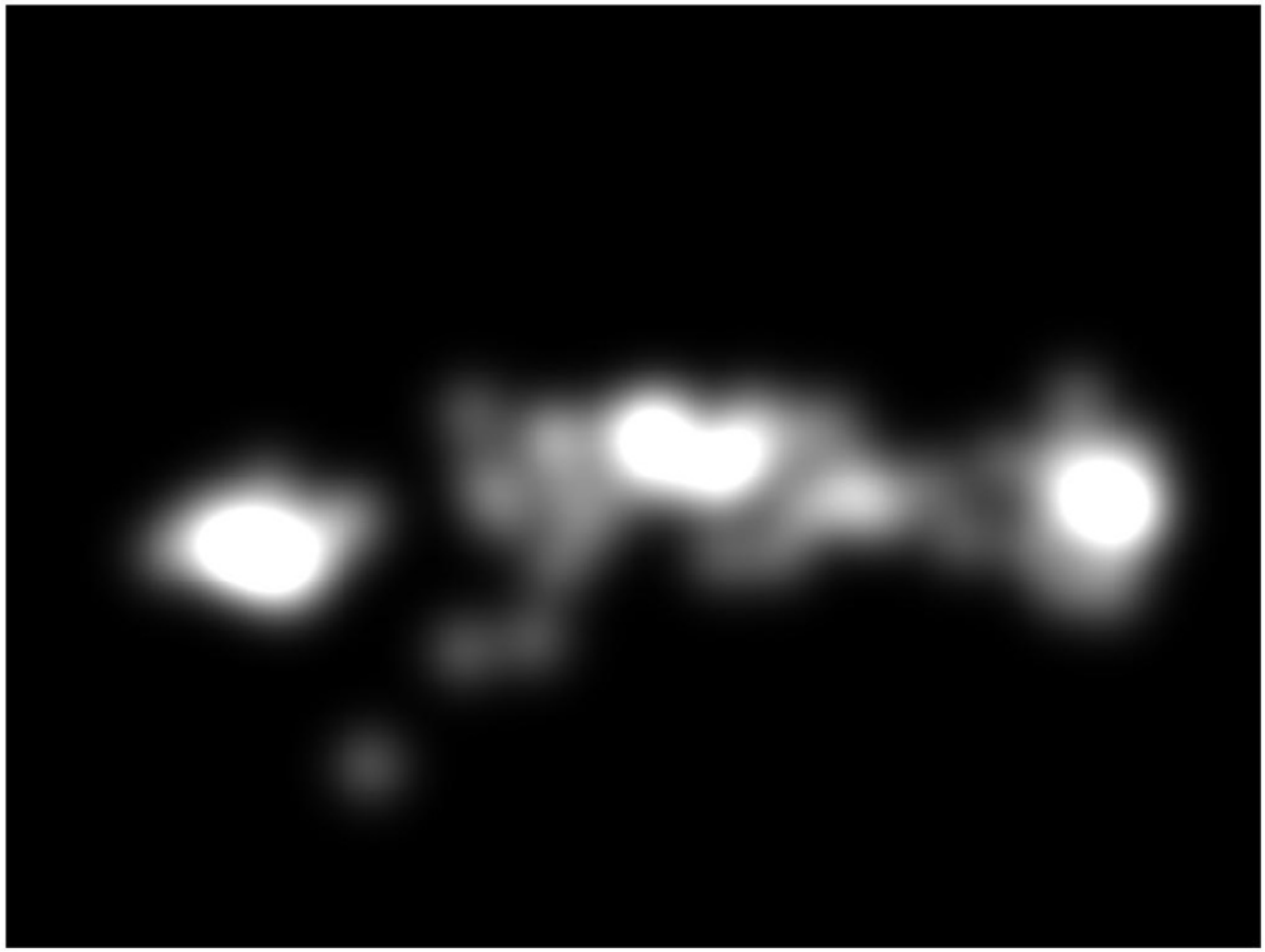
Fixations for One Person



Fixations
Vs
Saccades

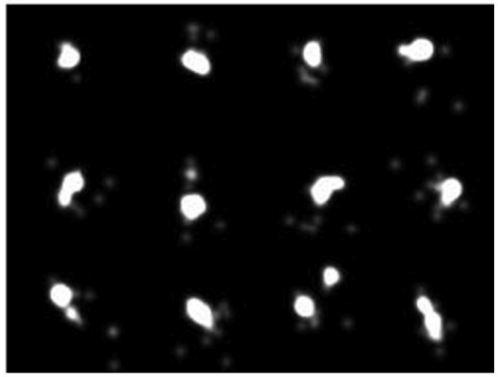
Fixations from 15 People

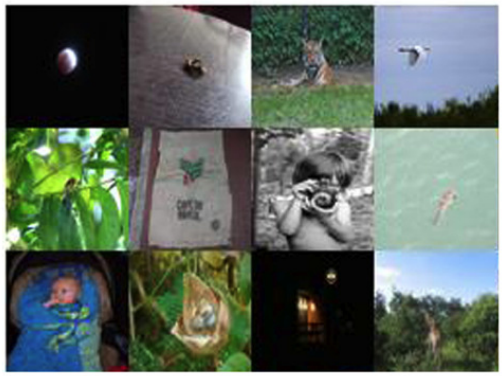
Fixation Map



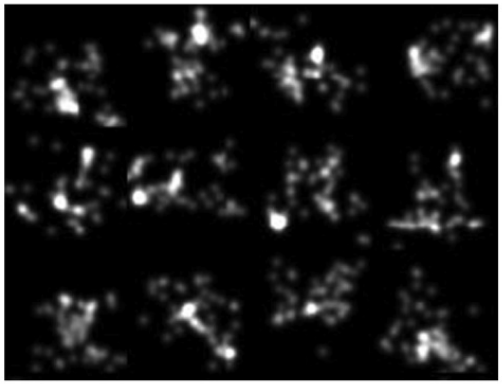
- Just convolve a Gaussian over the fixation positions
- Can do this for a single person, or an aggregate of many people
- Can also threshold:
  - Choose top n %
  - Yields a binary map
  - See Figure 1D

Analysis: Fixation Consistency Depends on Image Content

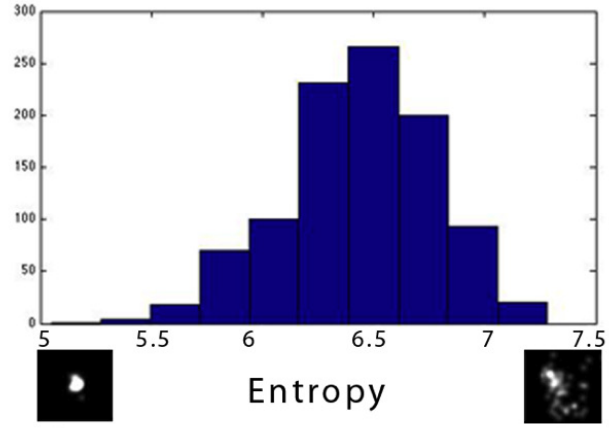


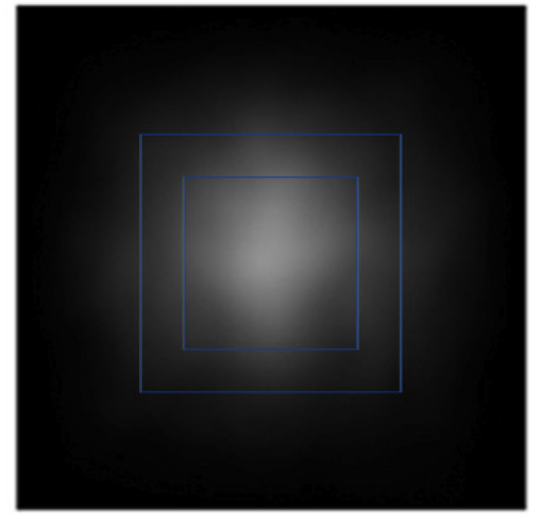


Low entropy saliency maps



High entropy saliency maps





Avg of all saliency maps

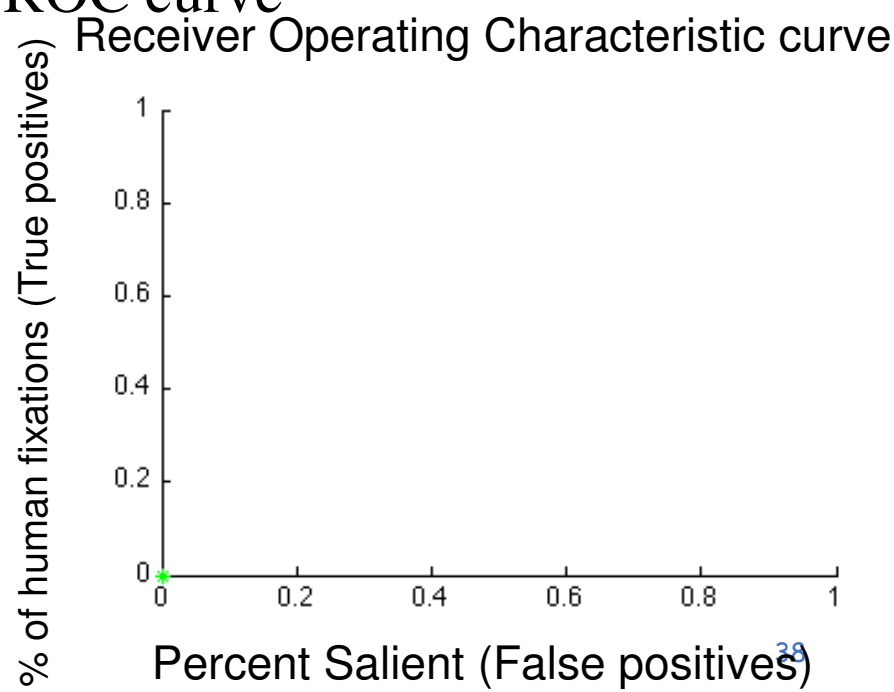
Observation: Fixations are Center Biased

- 40% of fixations are within the center 11% of the image
- 70% of fixations lie within the center 25% of the image

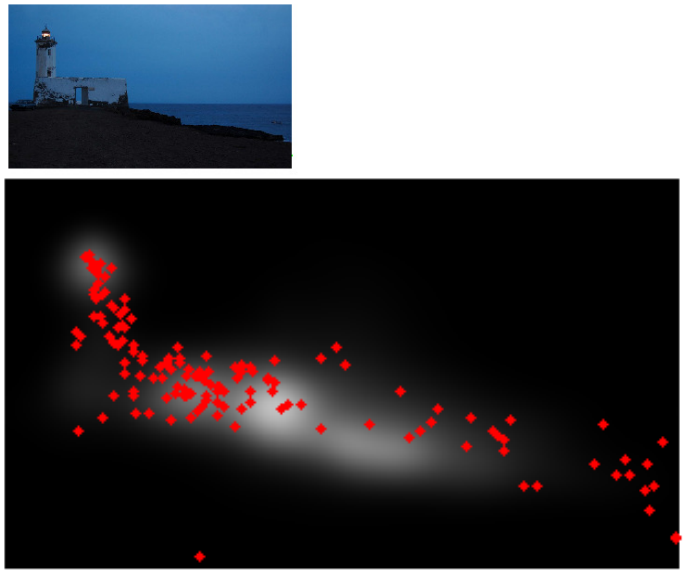
- Why?
  - **Photographic Bias** – Photographer puts objects of interest near center
  - **Viewing Bias** – People expect to see the objects of interest near the center
  - **Experiment Bias** – Viewers sitting right in front of the screen (not angled)
  - **Viewing Strategy** – Center is a good place to start exploring
  - **Orbital Reserve** – The eyes tend to look straight ahead (muscle laziness)

Evaluating Saliency Maps

- Saliency map from fixations of one person is thresholded to become a binary classifier
  - 1 if pixel at x,y over threshold, 0 otherwise
  - Compare against fixations from the other 14 people
- By varying the threshold we can get a ROC curve
  - Receiver Operating Characteristic Curve
  - Average over all users and all images

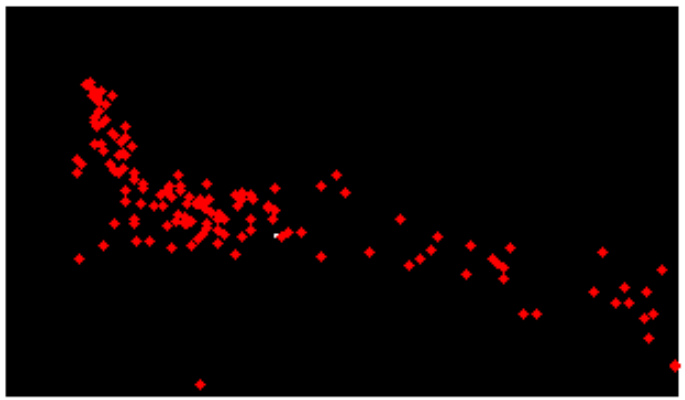


Human Fixations on Saliency Map

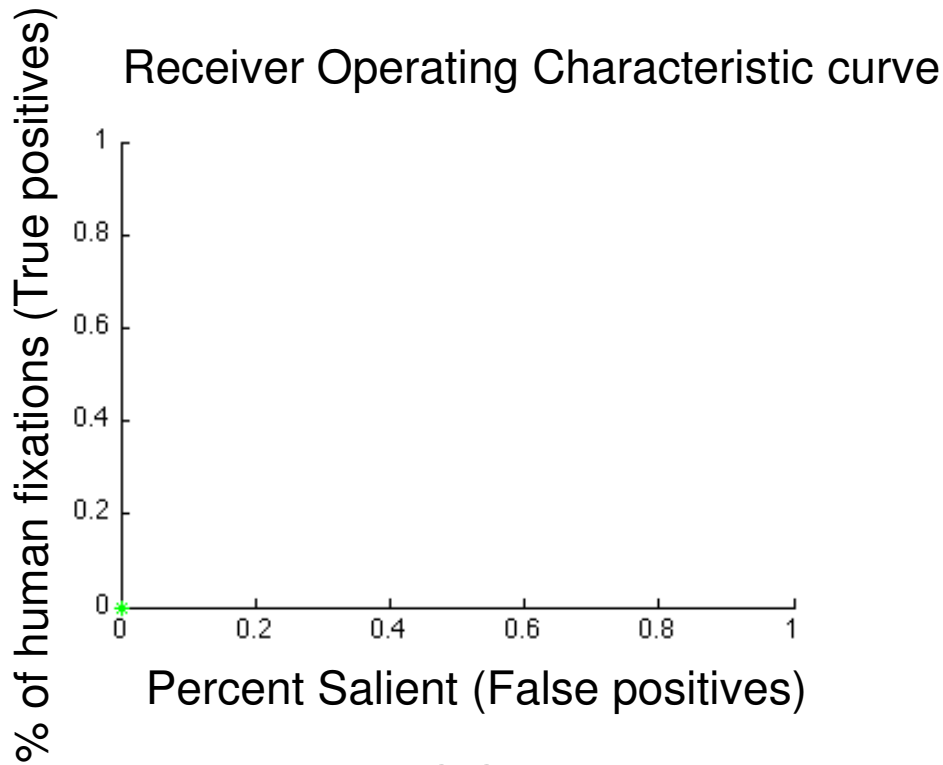


Human Fixations

ROC Curve - Humans

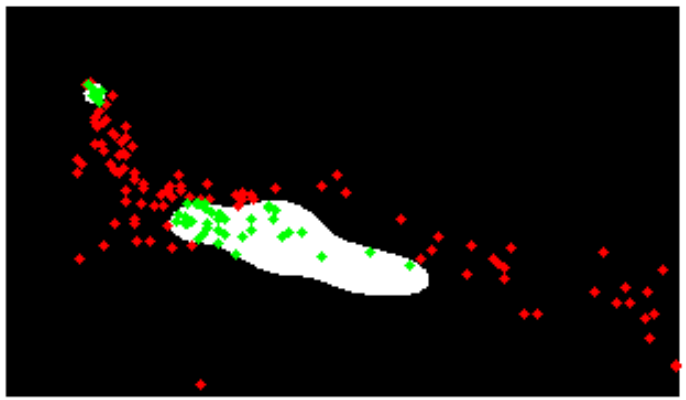


Thresholded Saliency Map

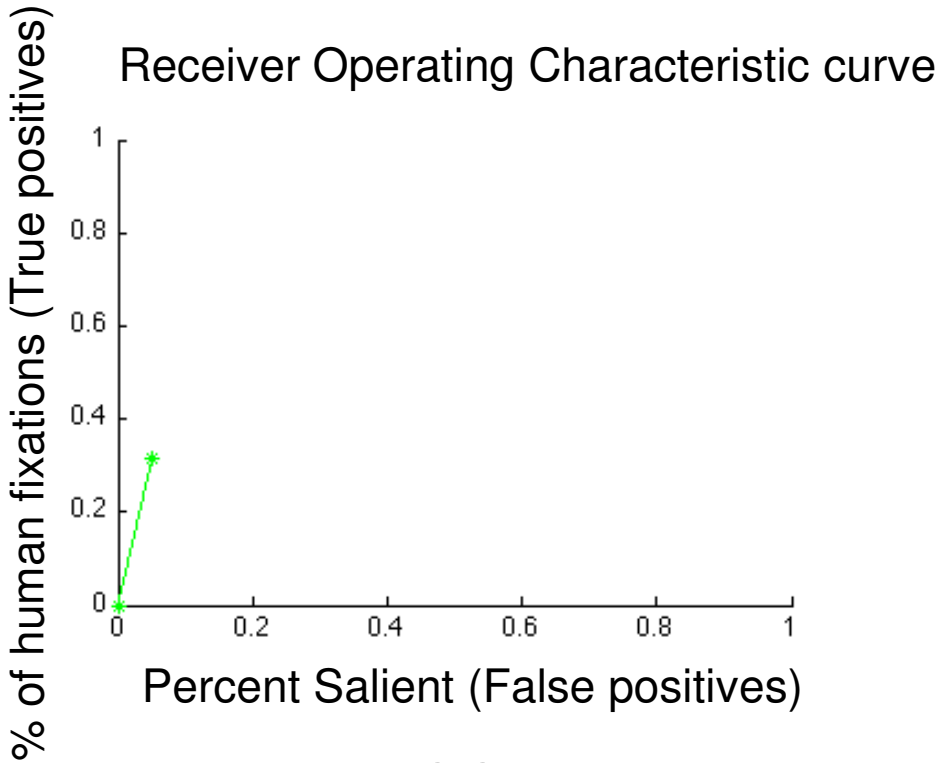


We calculate the percentage of fixations that lie within the salient portion of the map

ROC Curve – Humans

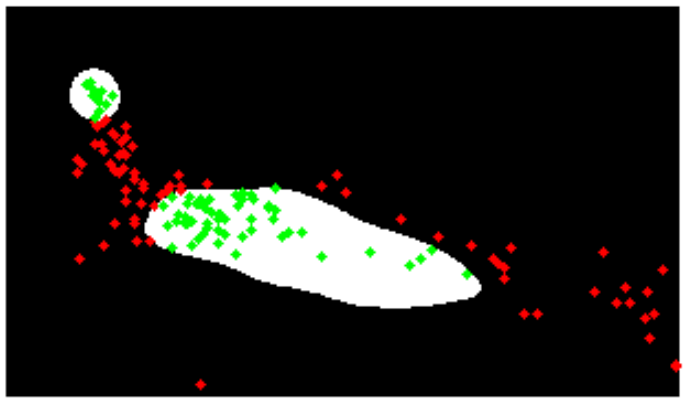


Thresholded Saliency Map

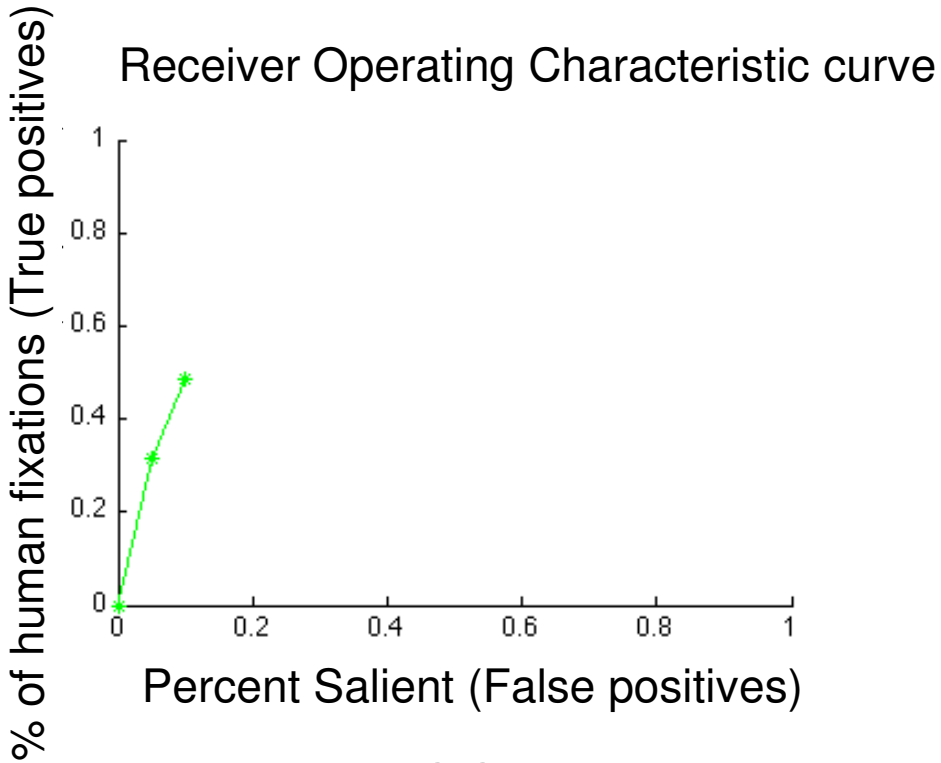


We calculate the percentage of fixations that lie within the salient portion of the map

ROC Curve - Humans

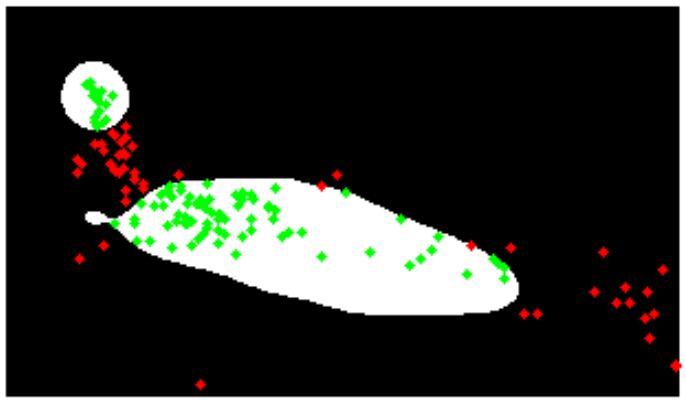


Thresholded Saliency Map

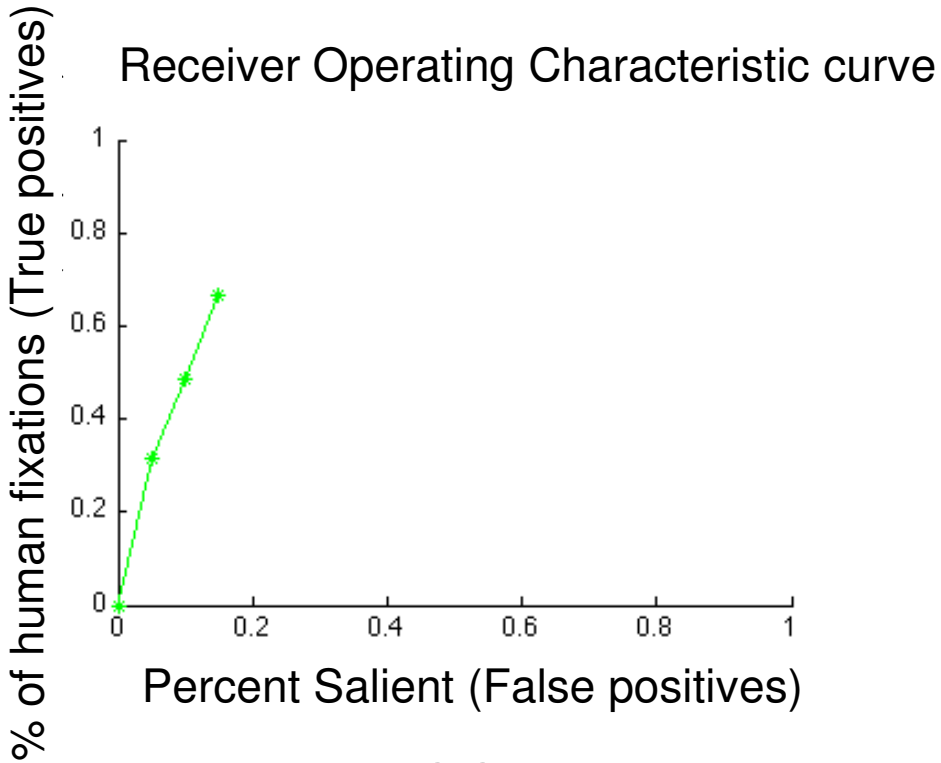


We calculate the percentage of fixations that lie within the salient portion of the map

ROC Curve

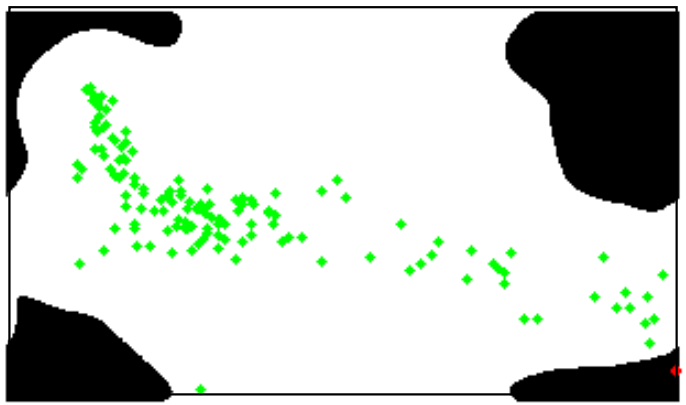


Thresholded Saliency Map

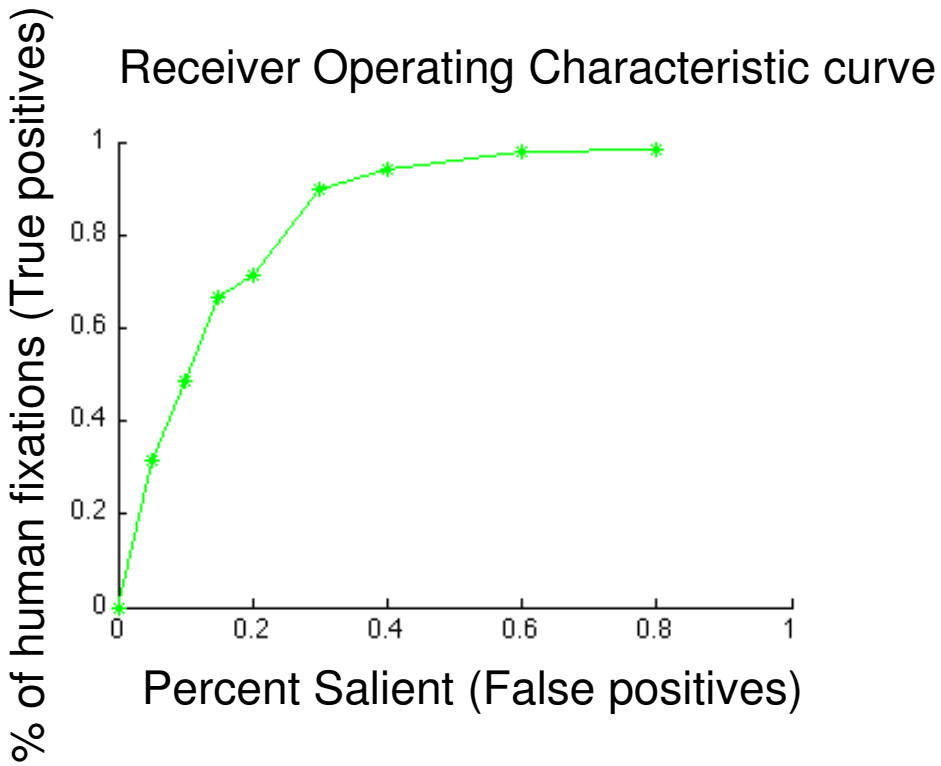


We calculate the percentage of fixations that lie within the salient portion of the map

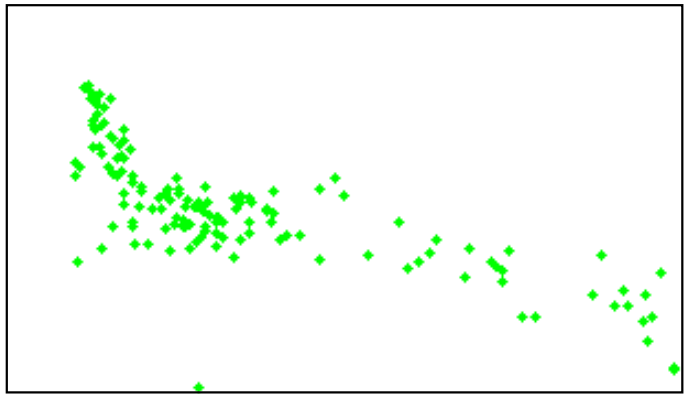
ROC Curve - Humans



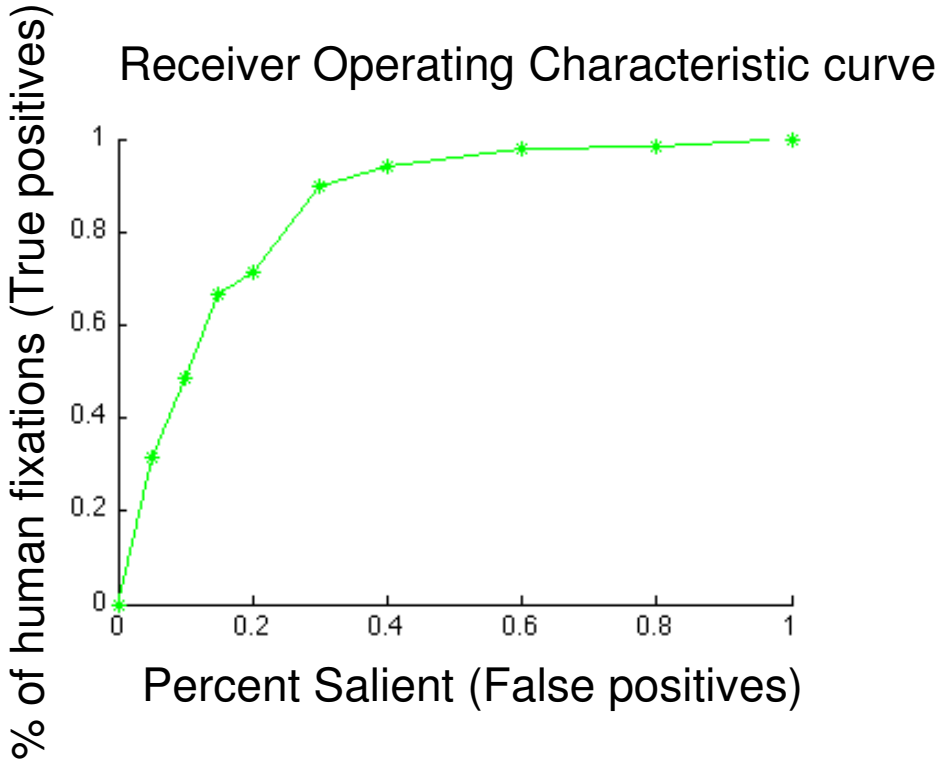
Thresholded Saliency Map



ROC Curve Humans

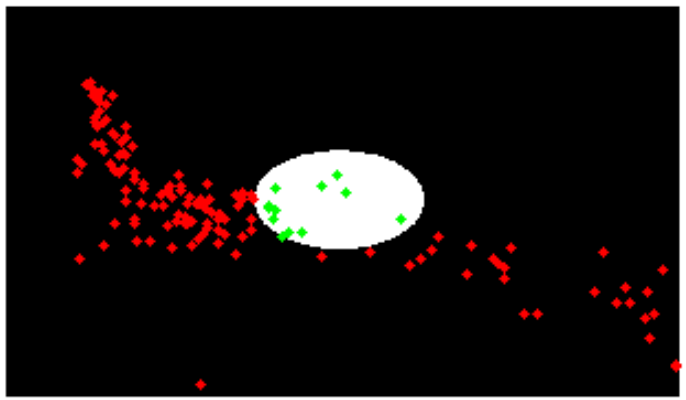


Thresholded Saliency Map

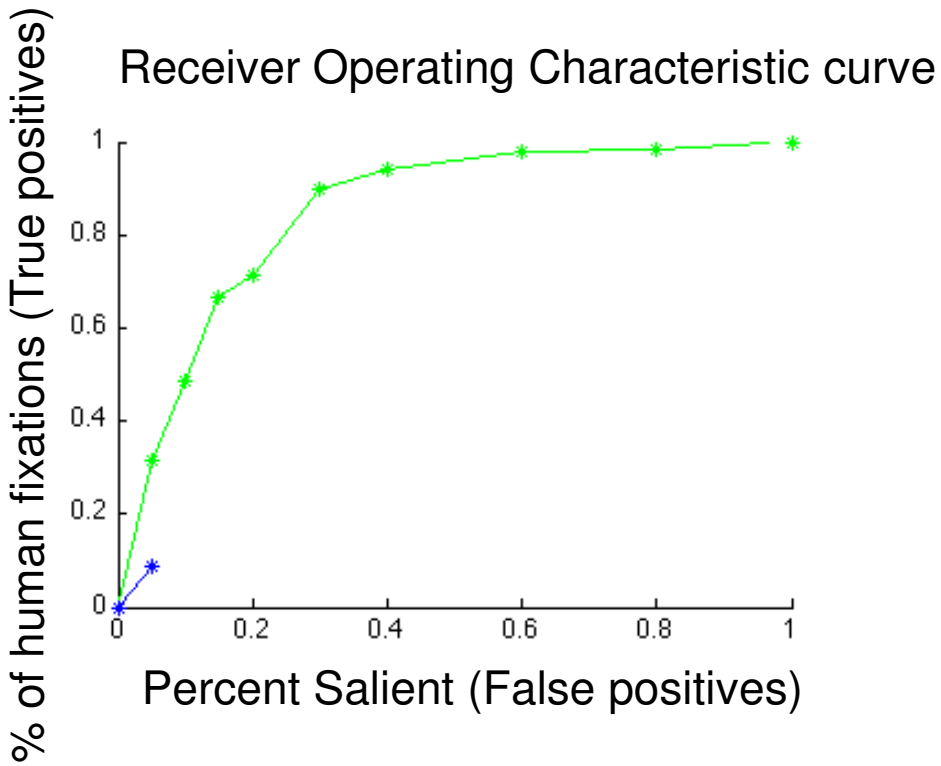


ROC curve always starts at 0
ends at 1

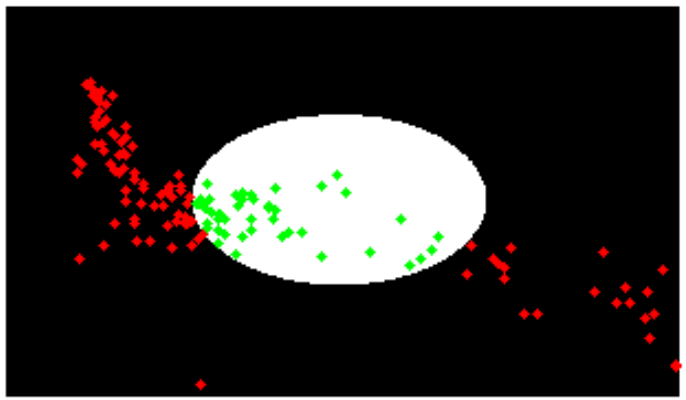
ROC Curve – Center Bias



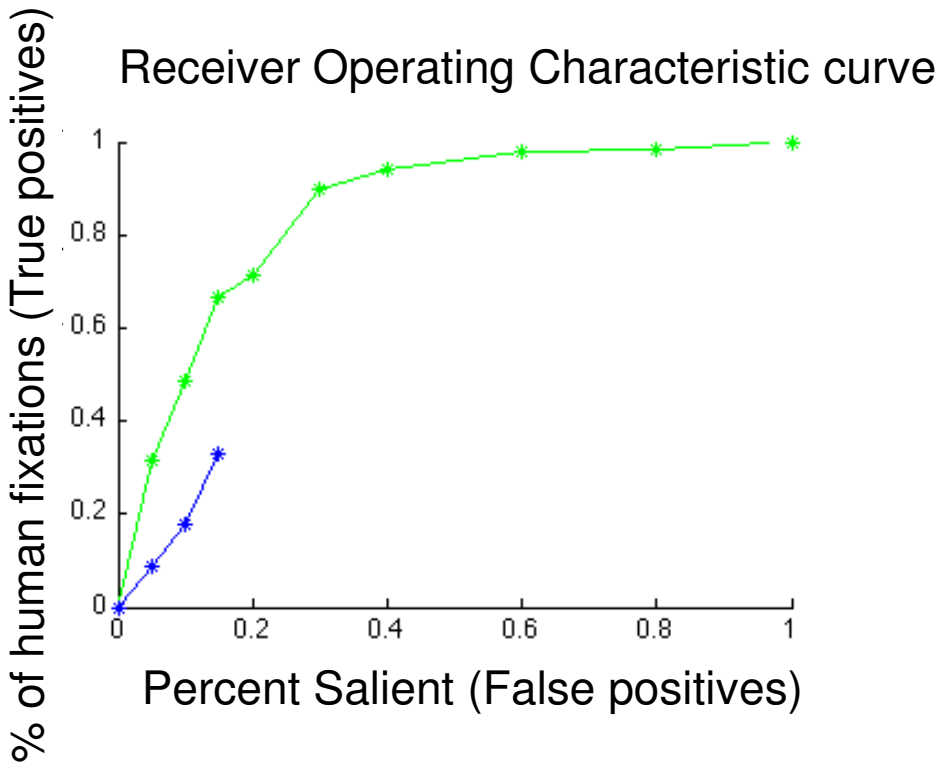
Thresholded Center Map



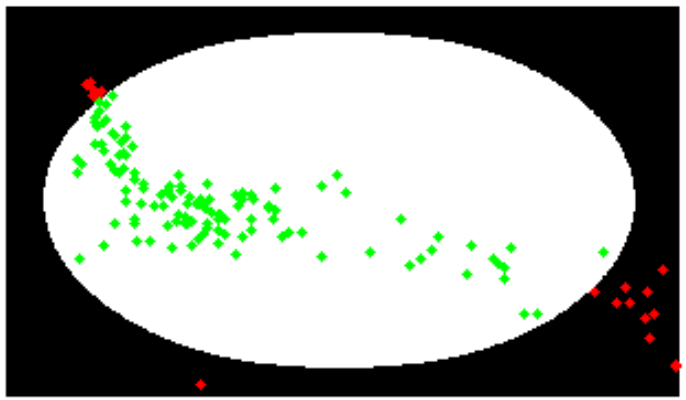
ROC Curve – Center Bias



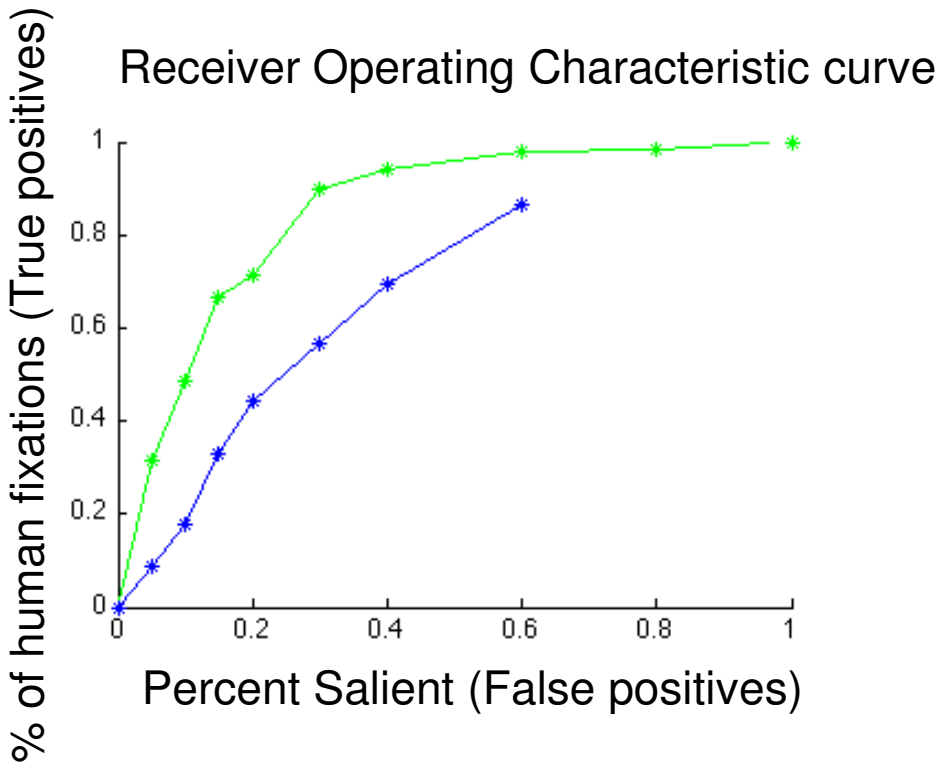
Thresholded Center Map



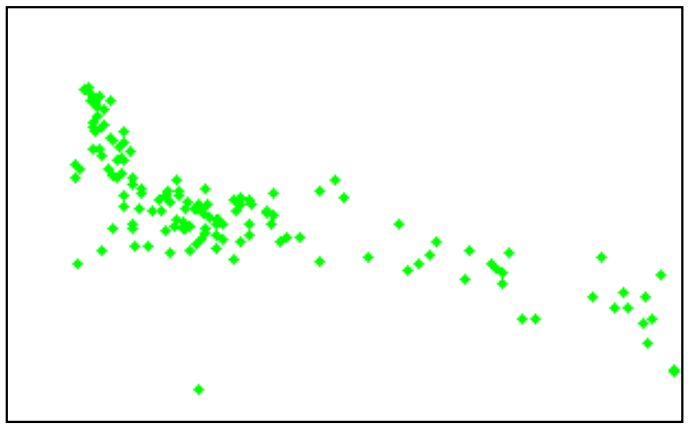
ROC Curve – Center Bias



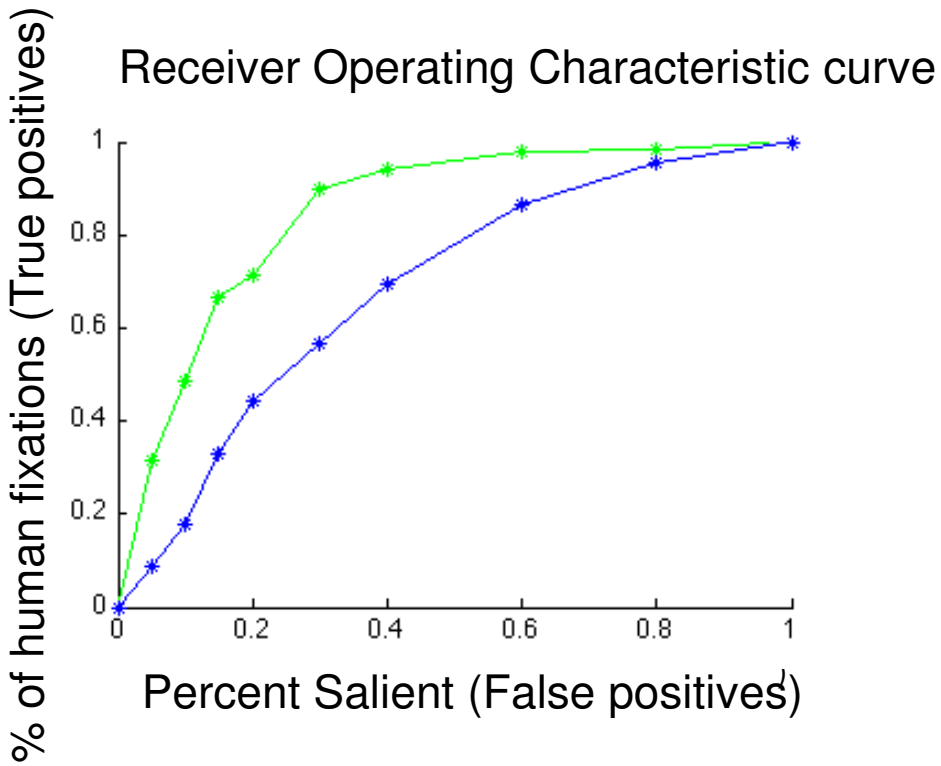
Thresholded Center Map



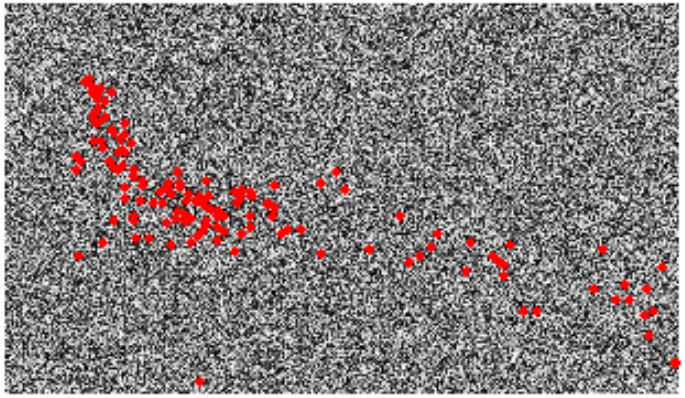
ROC Curve – Center Bias



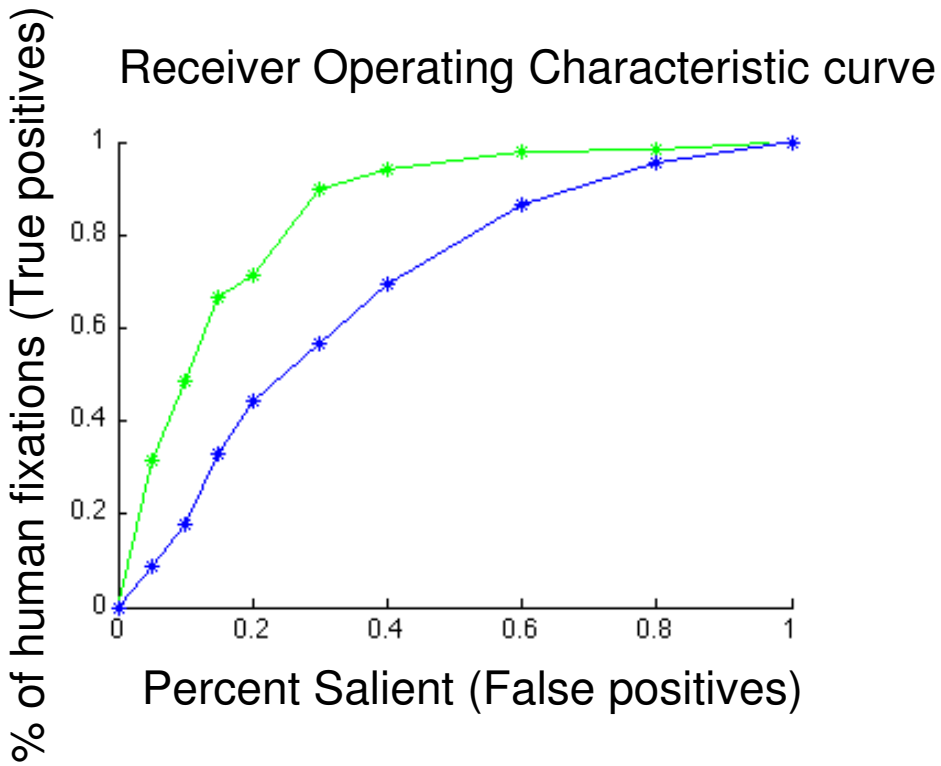
Thresholded Center Map



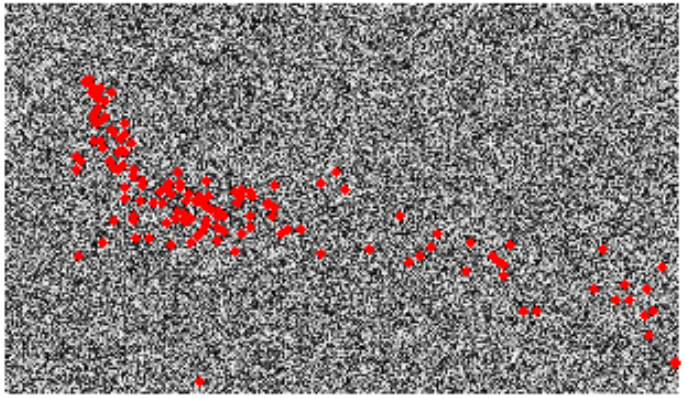
ROC Curve – Random Noise



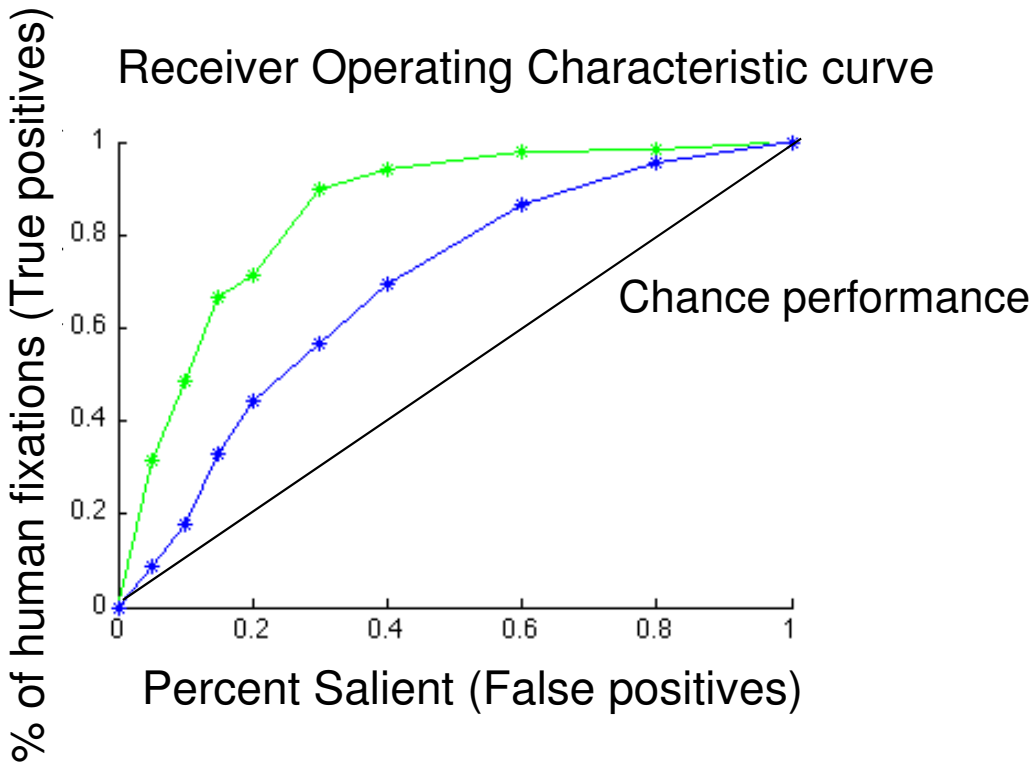
Random Noise map



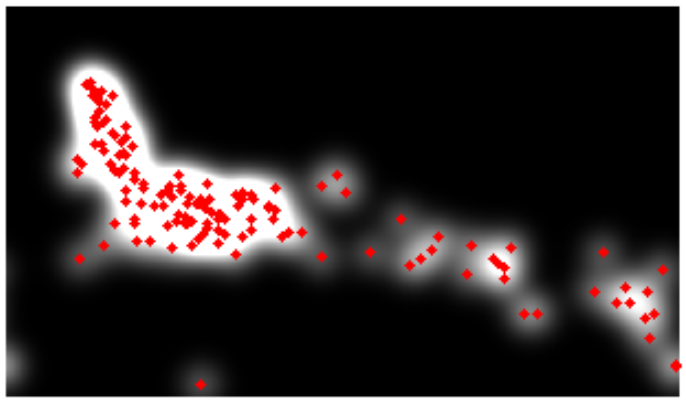
ROC Curve – Random Noise



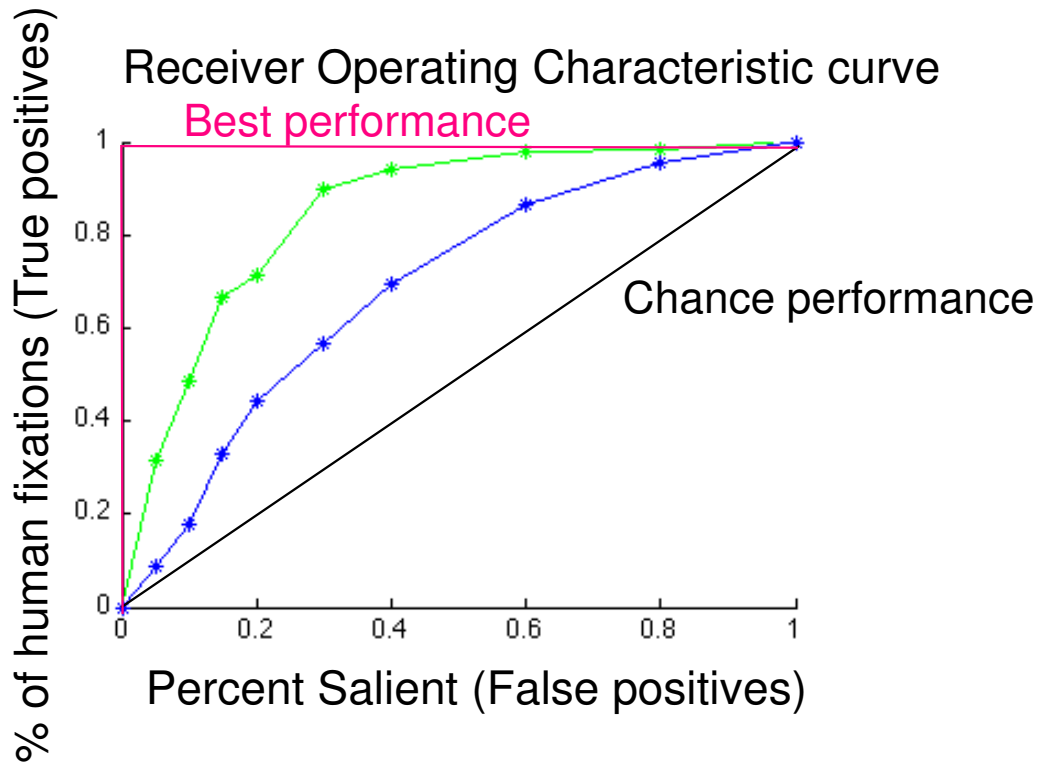
Random Noise map



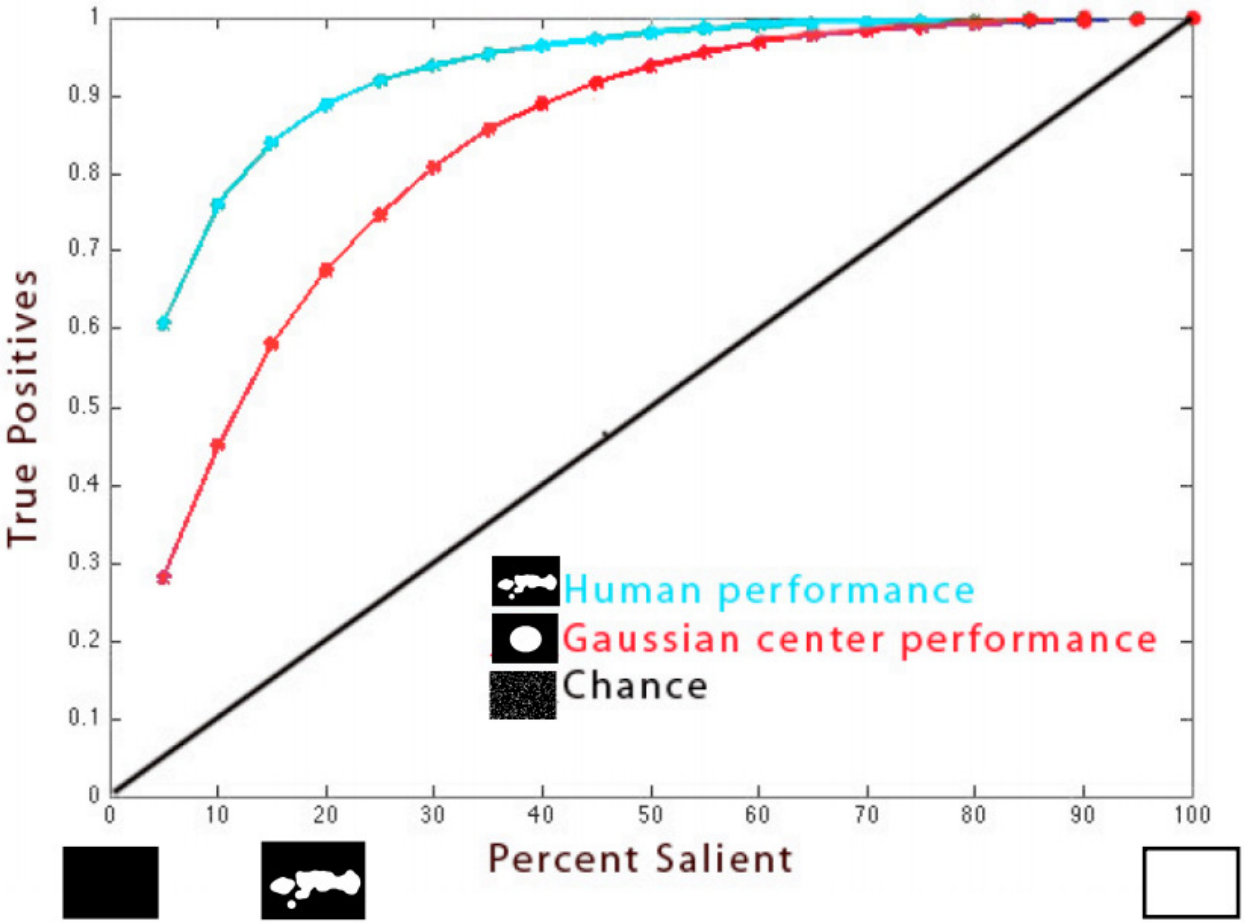
ROC Curve - Optimal

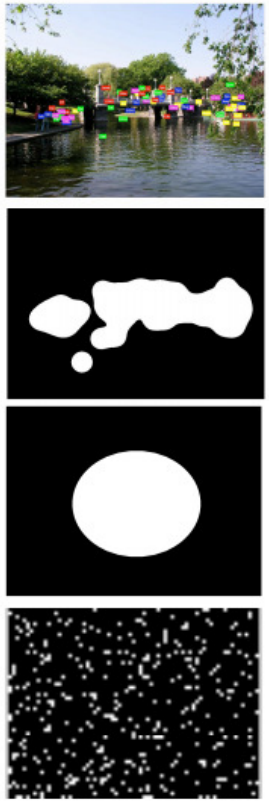


Perfect saliency map



ROC Curve in the Paper – Figure 5





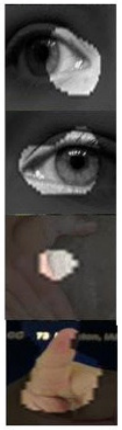
Where do people actually look?

- Labeled faces / text with bounding box and horizon line (if any)

Faces

People

Text



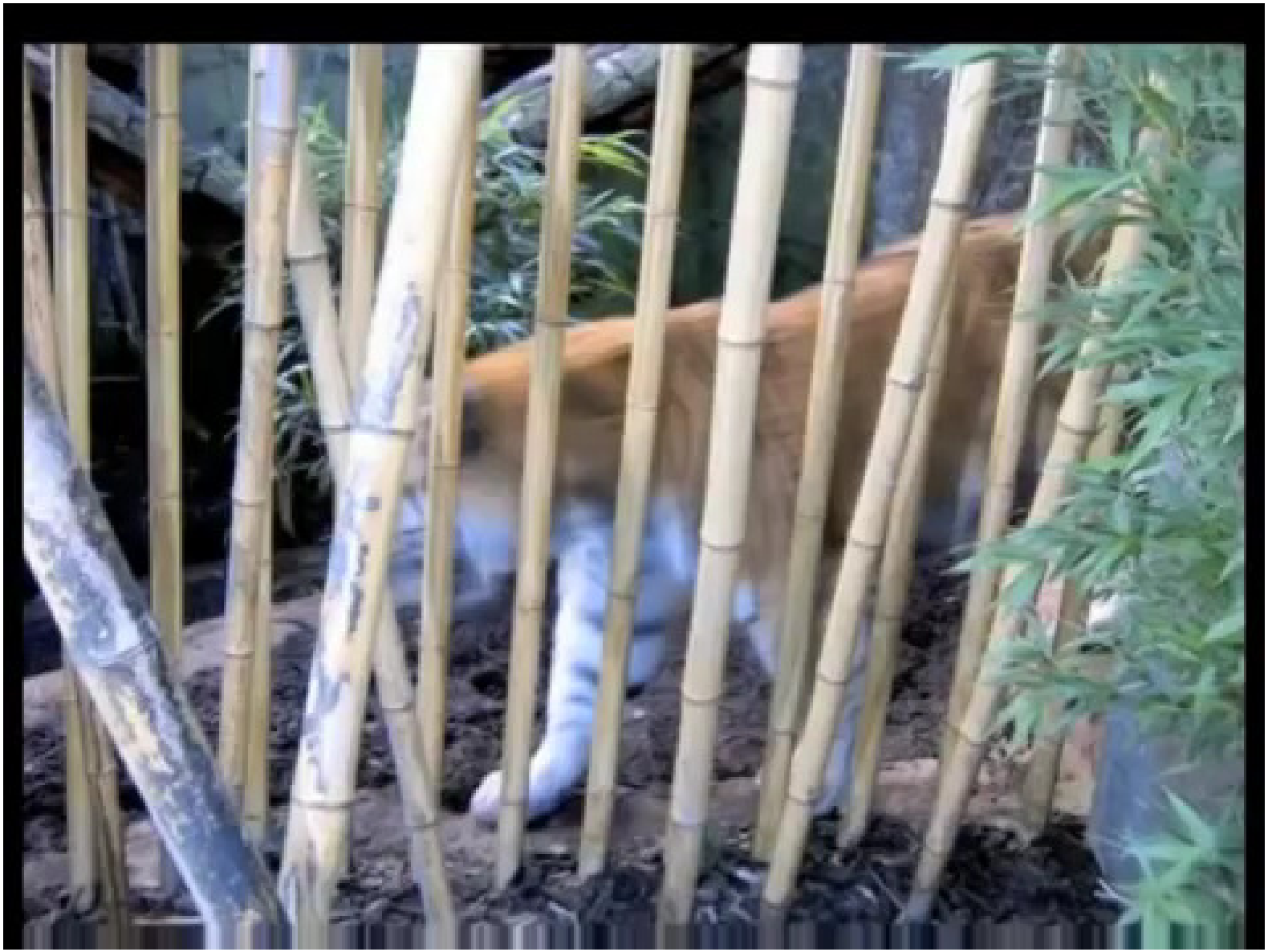
Body parts

Cars

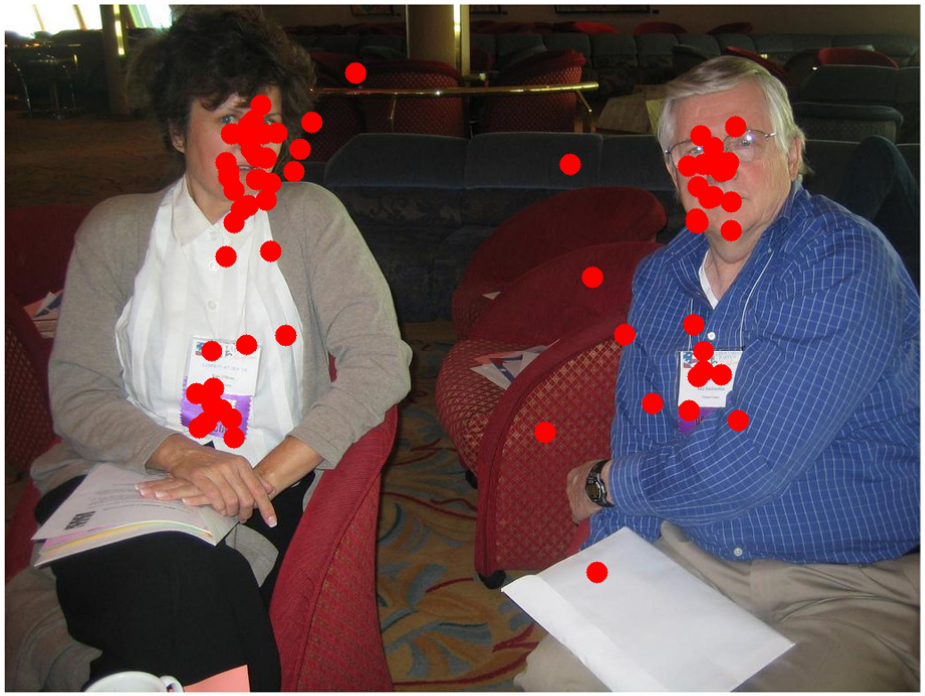
Animals

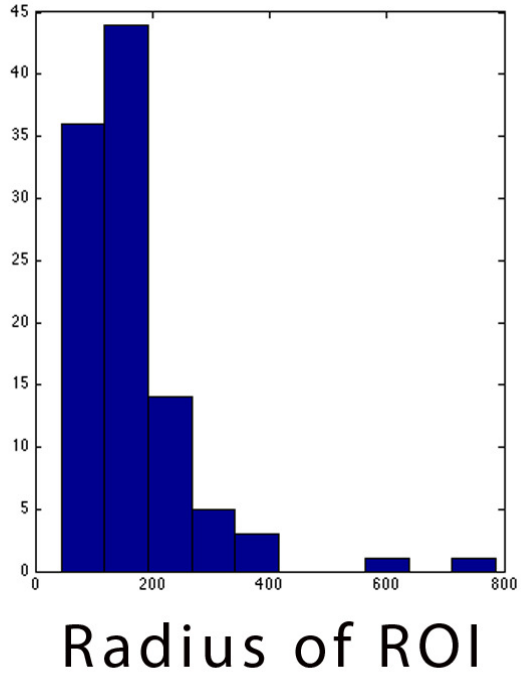
Based on the annotations:
10% fixation on faces
11% on text

Animals, cars, human body parts



Size of Regions of Interest





Drew a bounding box around clustered fixations and revealed that the radius of ROIs tend to be similar

Close-ups fixate on particular parts of the face, whereas in the other example focus on whole face

Learning Classifier from Eye-Tracking Data

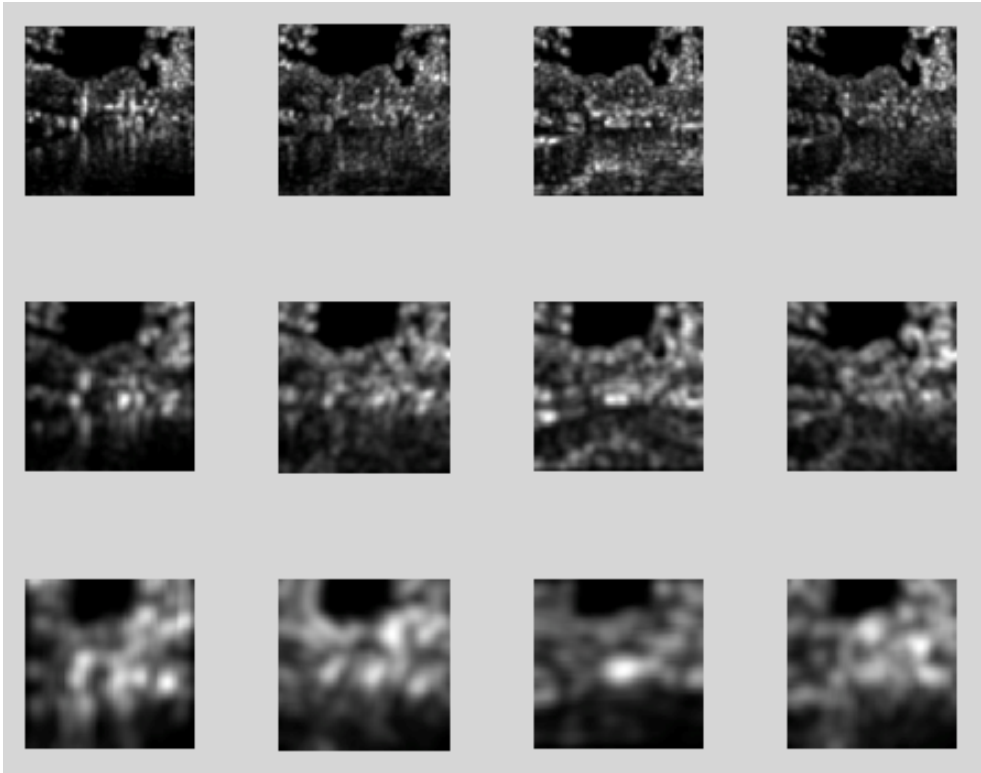
- Compute the features for a resized image of 200x200 to train

- Feature Types:
  - Low Level Features: Illumination, Orientation, Color…
    - Steerable pyramid filters, Itti-Koch, Torralba, RGB values and probabilities of each
  - Mid-Level Features: Horizon Line
    - Use gist features for this
  - High Level Features: Face detectors, person detectors
    - Viola-Jones face detector
    - DPM person detector
  - Center Prior – Distance to image center for each pixel

Low Level Image Features

Use subbands of the steerable pyramid filters (Simoncelli and Freeman 1995)
4 orientations

3 scales



Simple saliency map from Torralba based on steerable filters

Color Orientation Intensity

Itti-Koch

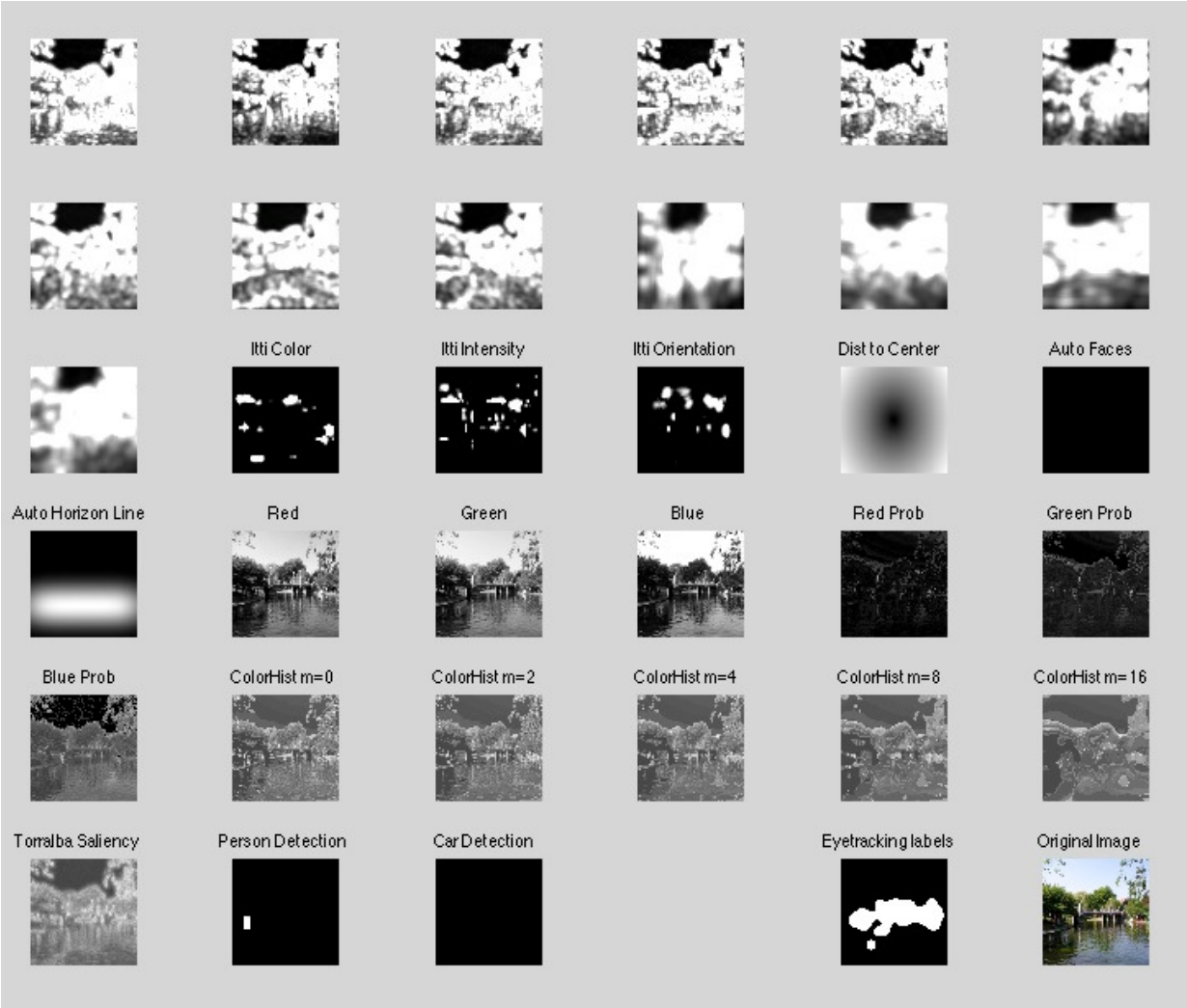
Low-Level Image Features (2)

Color channels

Probability of colors

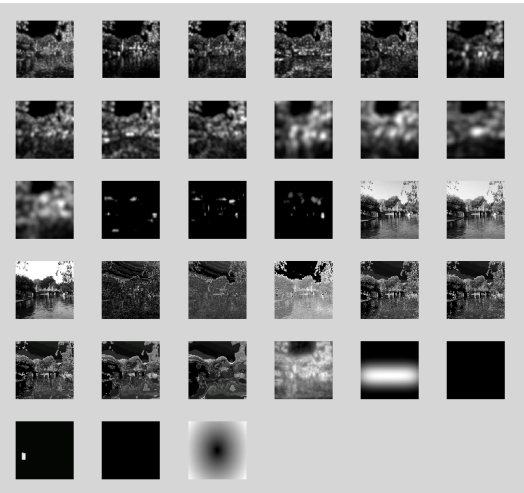
Compute probability of colors from 3D color histograms at different scales

All Features (2)



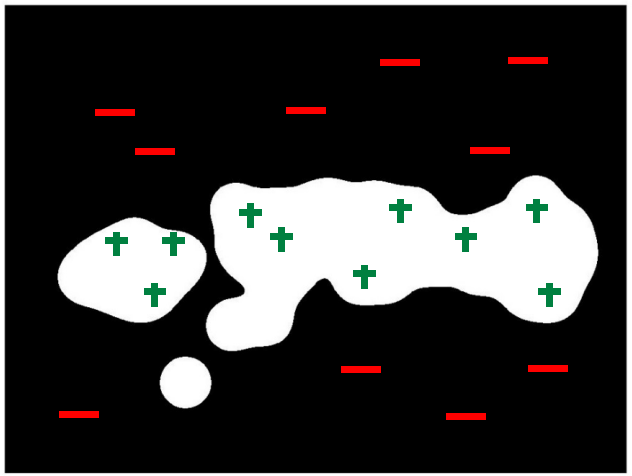
Training Procedure

33 features



+

Labels



10 salient samples from top 20%

10 non-salient samples from bottom 70%

from each of 903 training images

→ Support Vector Machine → **Model**

learns weights for each feature that best predict labels

Use linear kernels

Testing / Training Details

- Training:
  - Observations:
    - Adding more than 10 samples did not improve the performance
    - Using different ratios of positive / negative did not affect performance
  - Features were normalized to zero mean and unit variance before training
  - 9030 positive and 9030 negative training instances
  - Train models for each one of the types of features to compare against each other
- Testing:
  - For each test image, compute the 33 features.
    - For every pixel in the image
      - ✓ Give SVM the features at that pixel location from all the feature maps
      - ✓ Directly use the SVM score as the saliency per pixel (don't just do a binary classification)
    - Threshold the image and compute ROC scores
  - Test set size 100

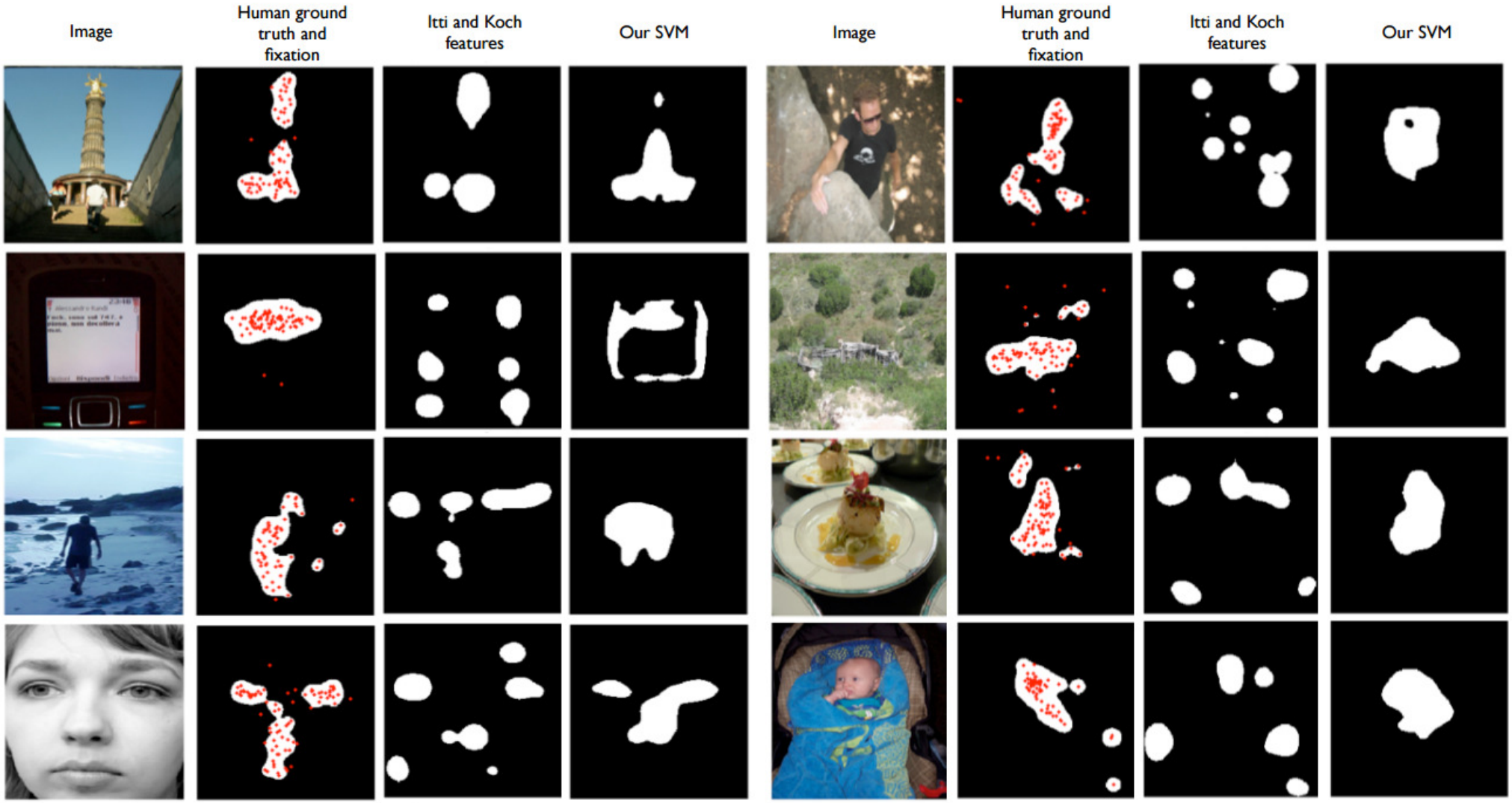
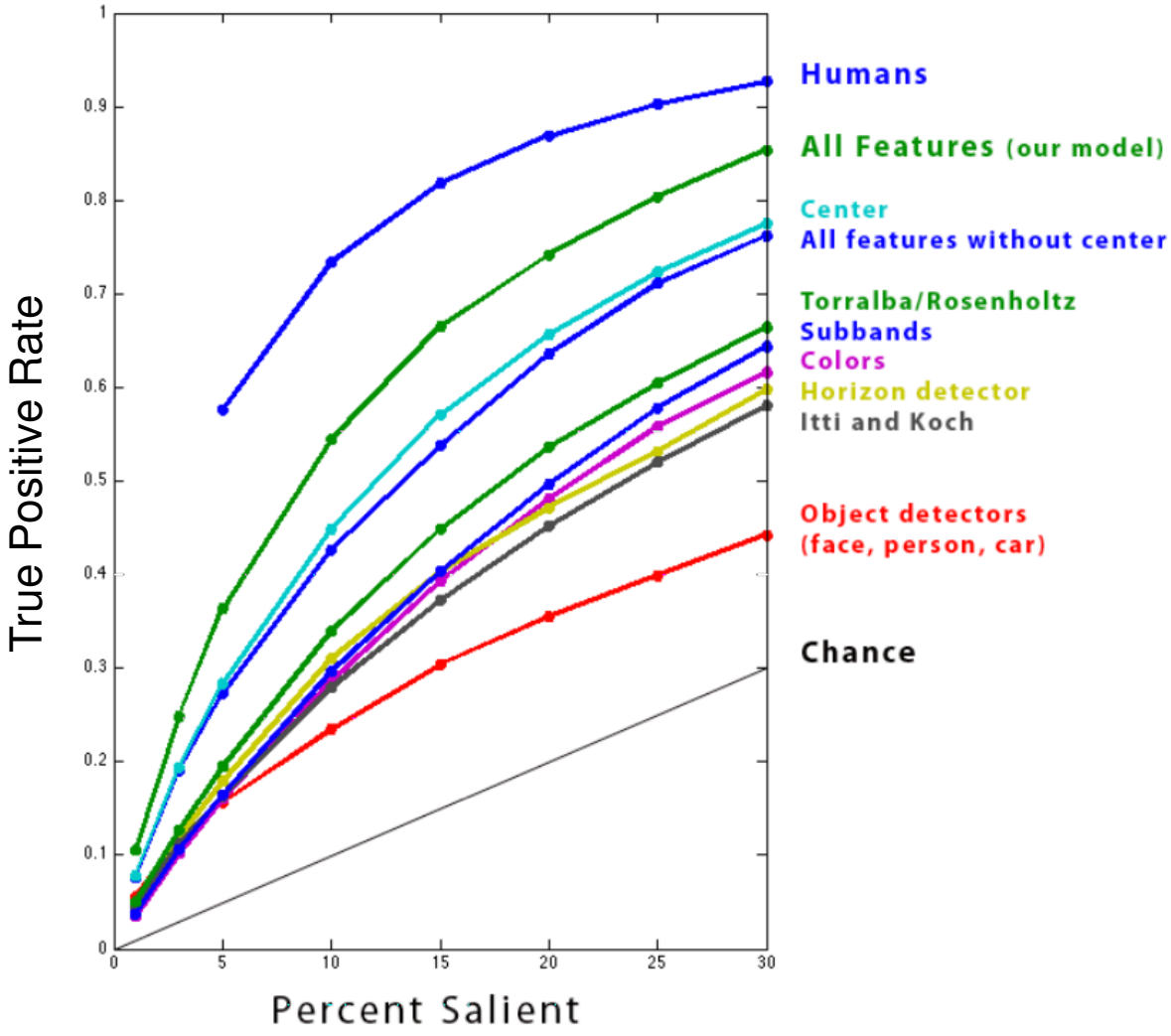


Image
Human ground truth and fixation
Itti and Koch features
Our SVM
Image
Human ground truth and fixation
Itti and Koch features
Our SVM


True Positive Rate
Percent Salient
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
Humans
All Features (our model)
Center
All features without center
Torralba/Rosenholtz
Subbands
Colors
Horizon detector
Itti and Koch
Object detectors
(face, person, car)
Chance

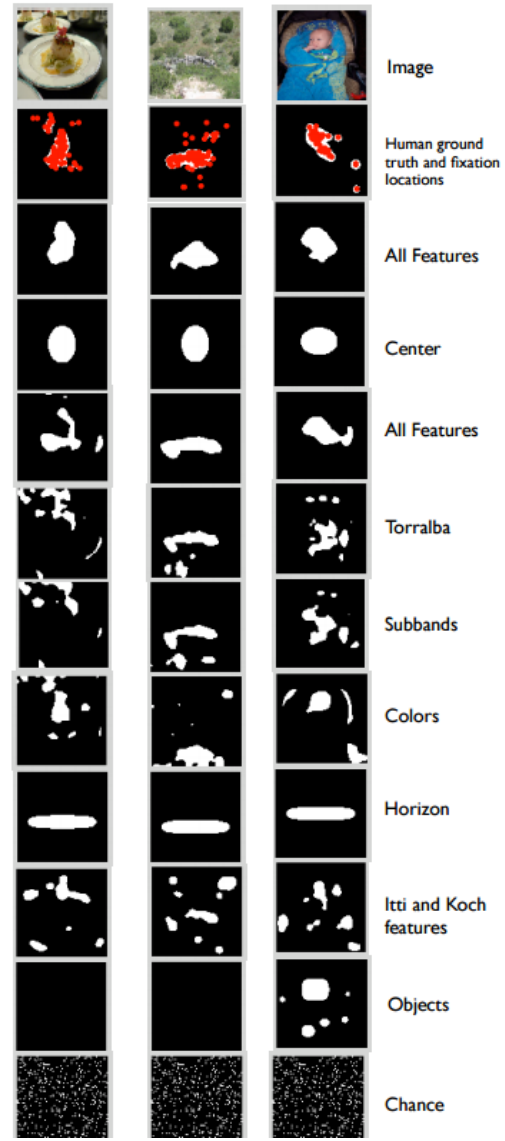
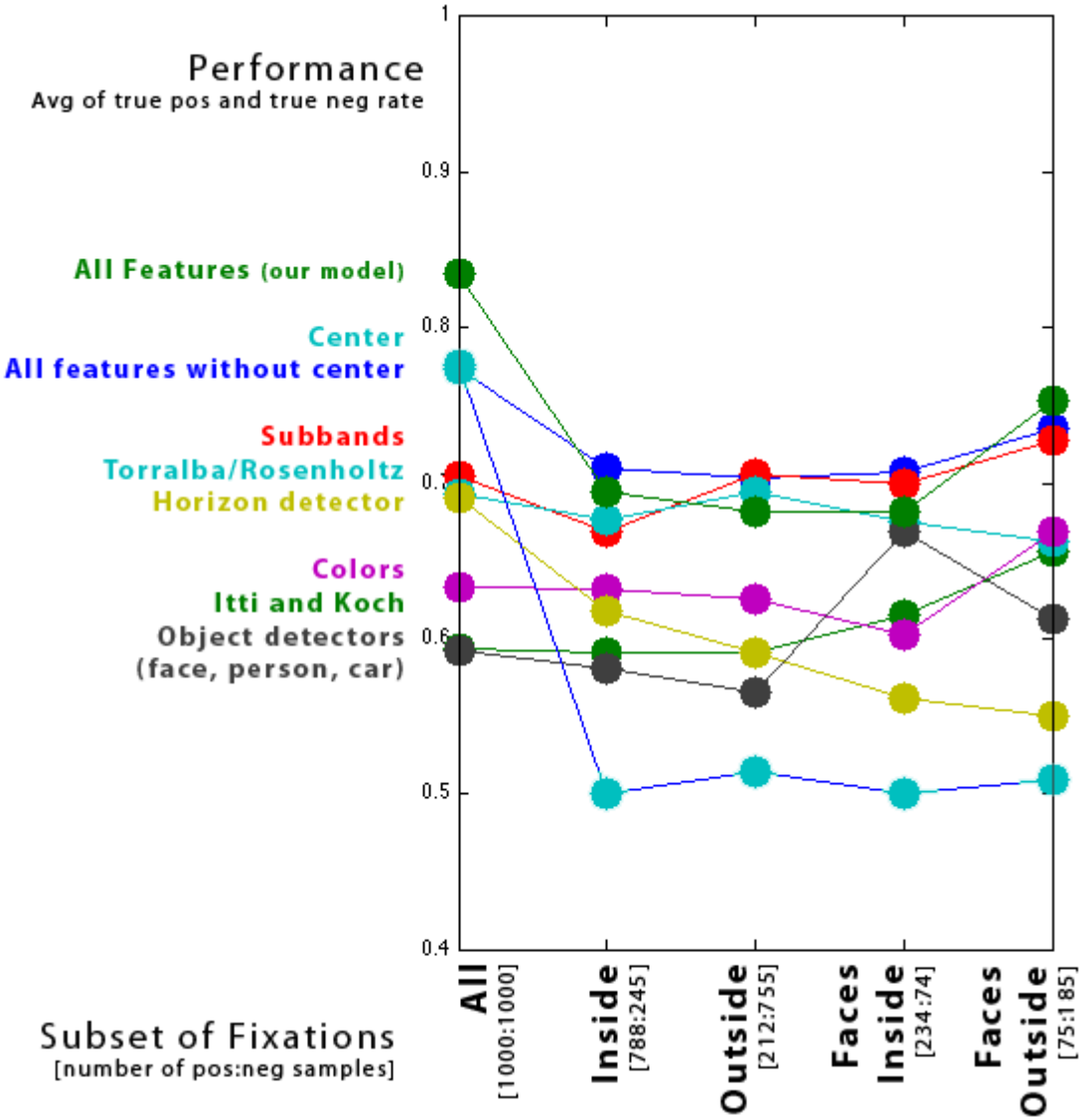
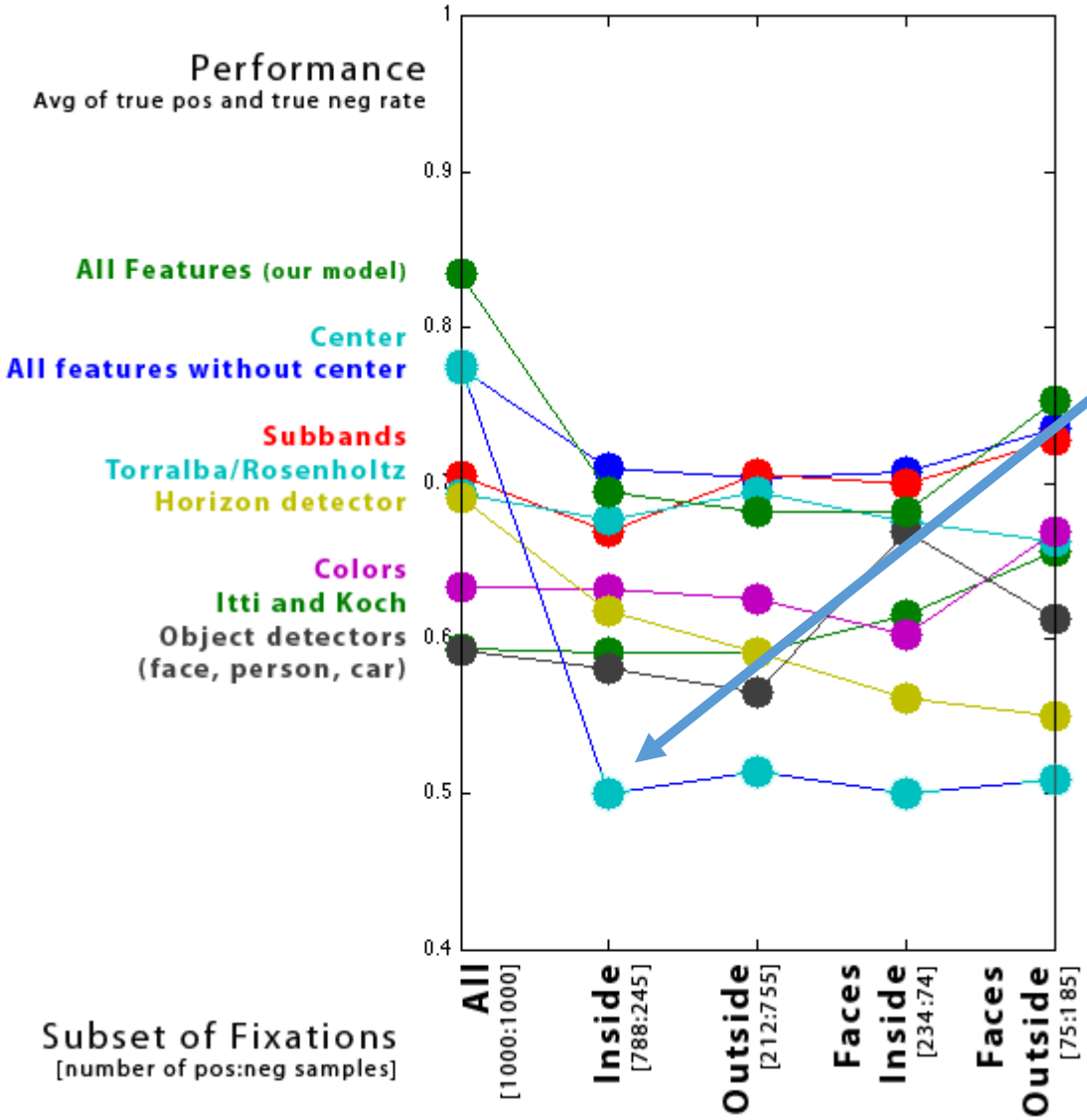


Image
Human ground truth and fixation locations
All Features
Center
All Features
Torralba
Subbands
Colors
Horizon
Itti and Koch features
Objects
Chance

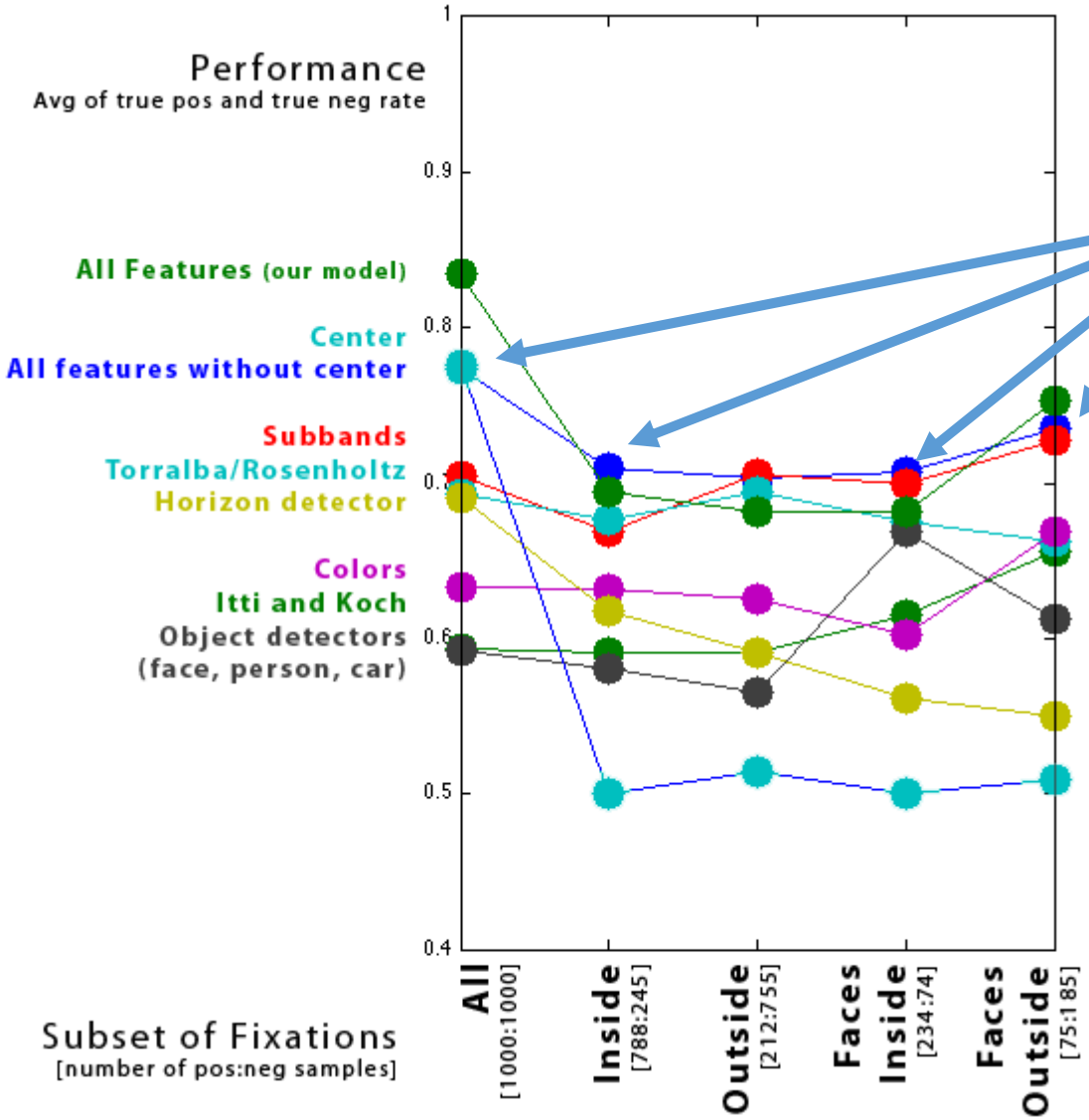
Second Experiment

- Motivation – Want to understand the impact of the center bias
- Experiment Design:
  - Design the image into a circular central region surrounded by a peripheral region
  - Central region is 0.42 (normalized) units from center or less
    - This region was labelled positive and the rest was peripheral was negative
  - Divide the samples to consider those inside and outside of the center and test separately to see how good each model is at predicting samples in those regions
  - Also, use the face annotations from before to test how good the model is at finding samples inside face annotated regions and outside face annotated regions
- Report performance as the average of true positive and true negative rates

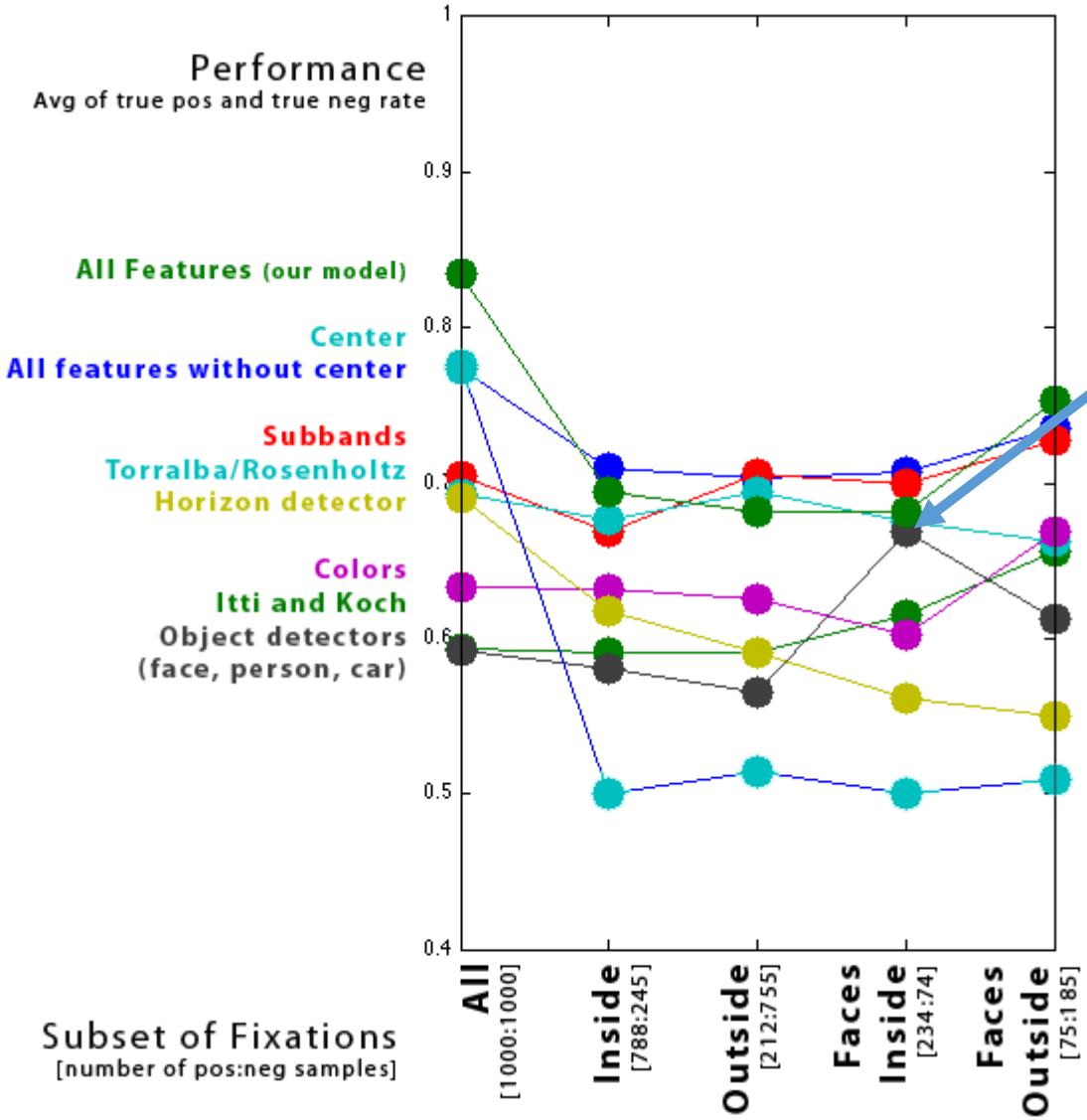

Performance
Avg of true pos and true neg rate
1
0.9
0.8
0.7
0.6
0.5
0.4
All Features (our model)
Center
All features without center
Subbands
Torralba/Rosenholtz
Horizon detector
Colors
Itti and Koch
Object detectors
(face, person, car)
All
[1000:1000]
Inside
[788:245]
Outside
[212:755]
Faces
Inside
[234:74]
Faces
Outside
[75:185]
Subset of Fixations
[number of pos:neg samples]



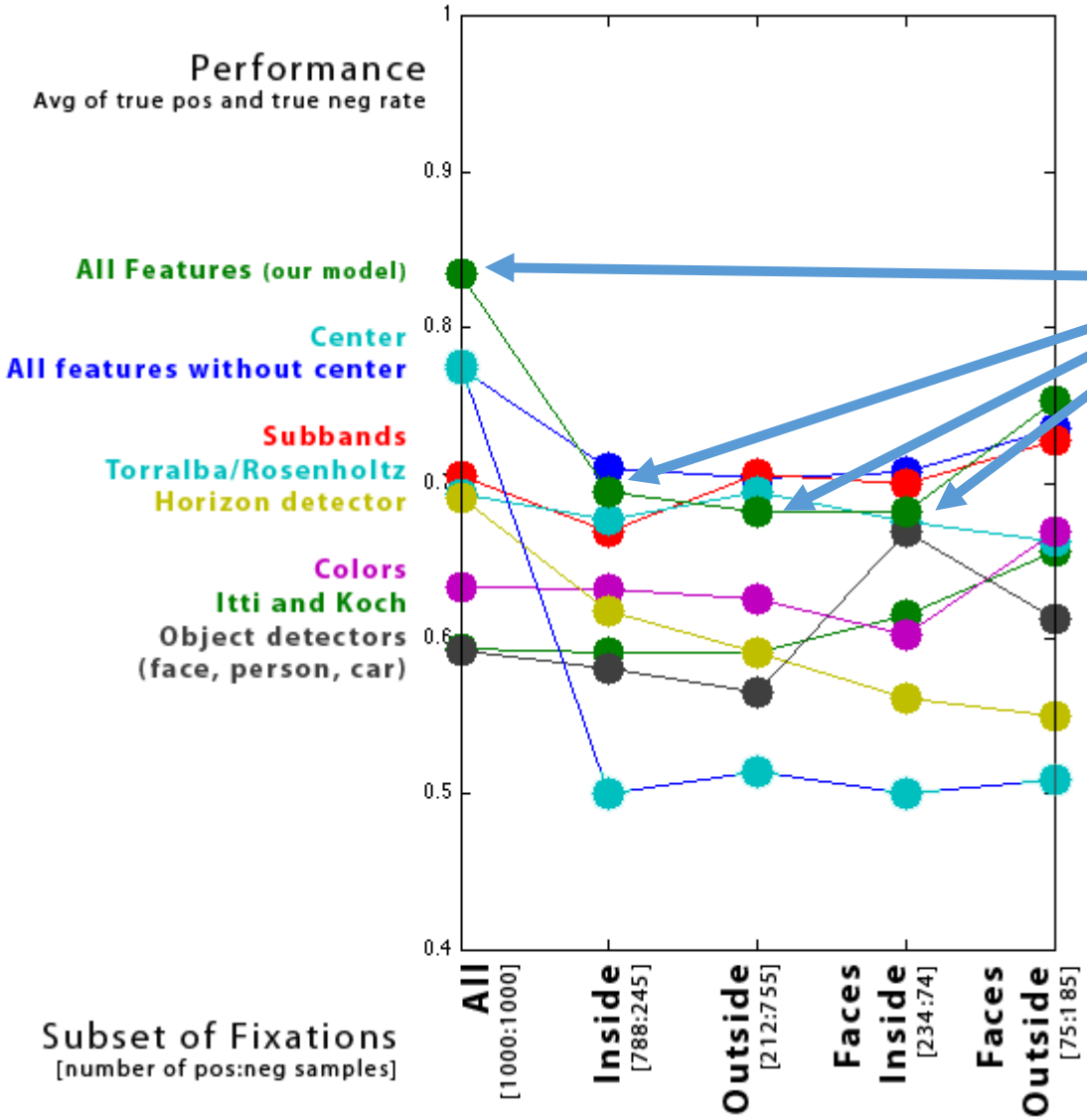
The center model performs only as well as chance for the subsets of samples



All features without center does roughly the same as center on the total set, but on the subsets, it is more robust



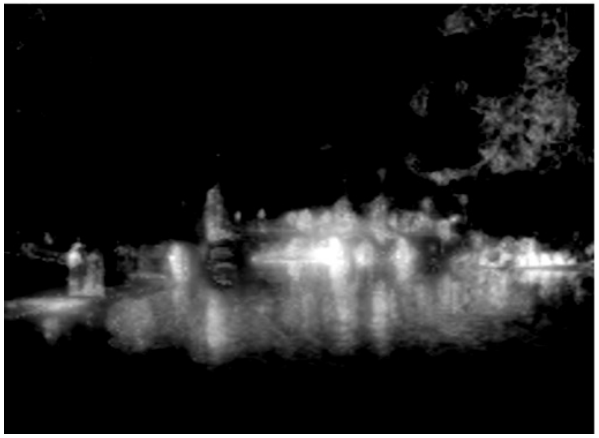
The object detector model performs best on the subsets with faces (as expected)



The All Features model (which uses the center prior) is outperformed on the subsets inside and outside

This is because it takes a hit because of the center bias, however all features without center remains roughly at the top for inside, outside, and faces inside, and almost for faces outside

NPR Application



Conclusions

- The paper presents an approach combining top-down and bottom-up cues to compute a saliency map of an image to determine where people are likely to look in the image
- The model is trained using human eye-tracking data and features fed into a SVM classifier to predict the saliency of each pixel
  - Could be considered a weakness → We need to have eye-tracking data
- Future Work:
  - Integrating context/scene – What objects are salient may depend on the context
  - How to integrate surprise? (objects which don't fit in that context)
  - Task?
  - Why not have a text detector? They know that text really matters so why not just detect text as another feature. This shouldn't be too difficult
  - Compare against truly top-down models of saliency
    - High level features here are detections from DPM but more robust Top down models exist
  - Objectness

Understanding and Predicting Importance In Images

Secondary Paper

Berg, et al.

Presented by Chris Thomas

Problem Overview

- Problem: What do people consider important in an image?
  - Different problem than where people look in an image

Problem Overview

- Problem: What do people consider important in an image?
  - Different problem than where people look in an image

"A raft with 3 adults and two children in a river."
"Four people in a canoe paddling in a river lined with cliffs."
"Several people in a canoe in the river."

Problem Overview

- Problem: What do people consider important in an image?
  - Different problem than where people look in an image

Why don't people describe this? Why isn't it important?

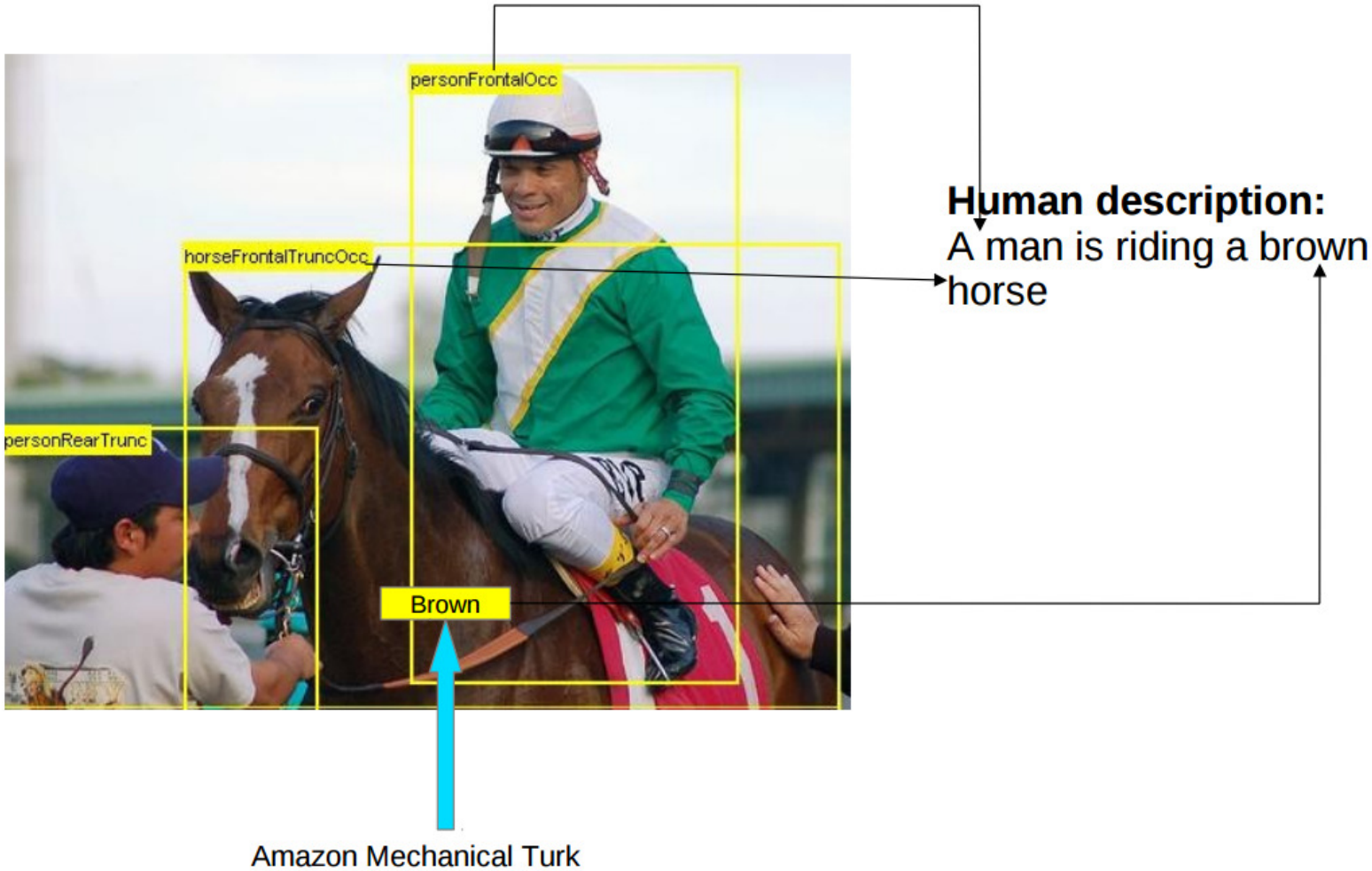
Paper Overview

- The authors propose a number of factors for predicting importance:
  - Compositional Factors –Size / location
  - Content Semantics – Object category (human vs rock) / scene category + strength
  - Contextual Factors – Object-Scene / Attribute-Object context
- They evaluate the models by looking at image content labels and descriptions of images written by people of images
- Learn models to predict what people would describe:
  - Given what is in the image (algorithm told what is there)
  - Automatically detect image content and then predict what is important
- **Motivations**: Image search / generating natural descriptions for images

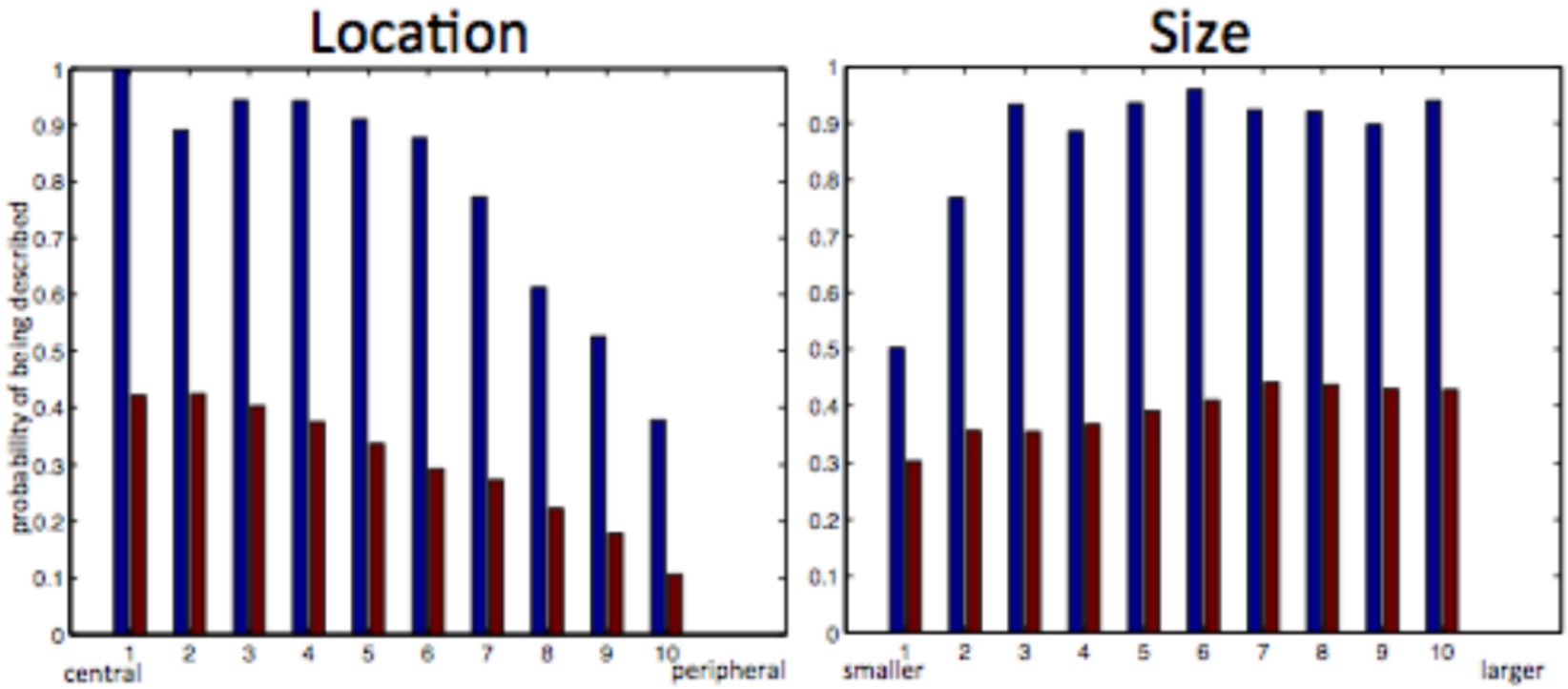
Approach Overview

- Collect Dataset
- Content Labels: Object labels, scene labels, attribute labels
  - Use Mturk to get scene labels and attribute labels
- Mapping from content to description
  - Need to know which object "child" or "bearded man" refers to
    - Also scene category labels + attribute labels
  - Perform this by hand
  - Also devise an automatic method based on matching description to label
- Evaluation of the factors
- Building / Evaluating predictive models of importance

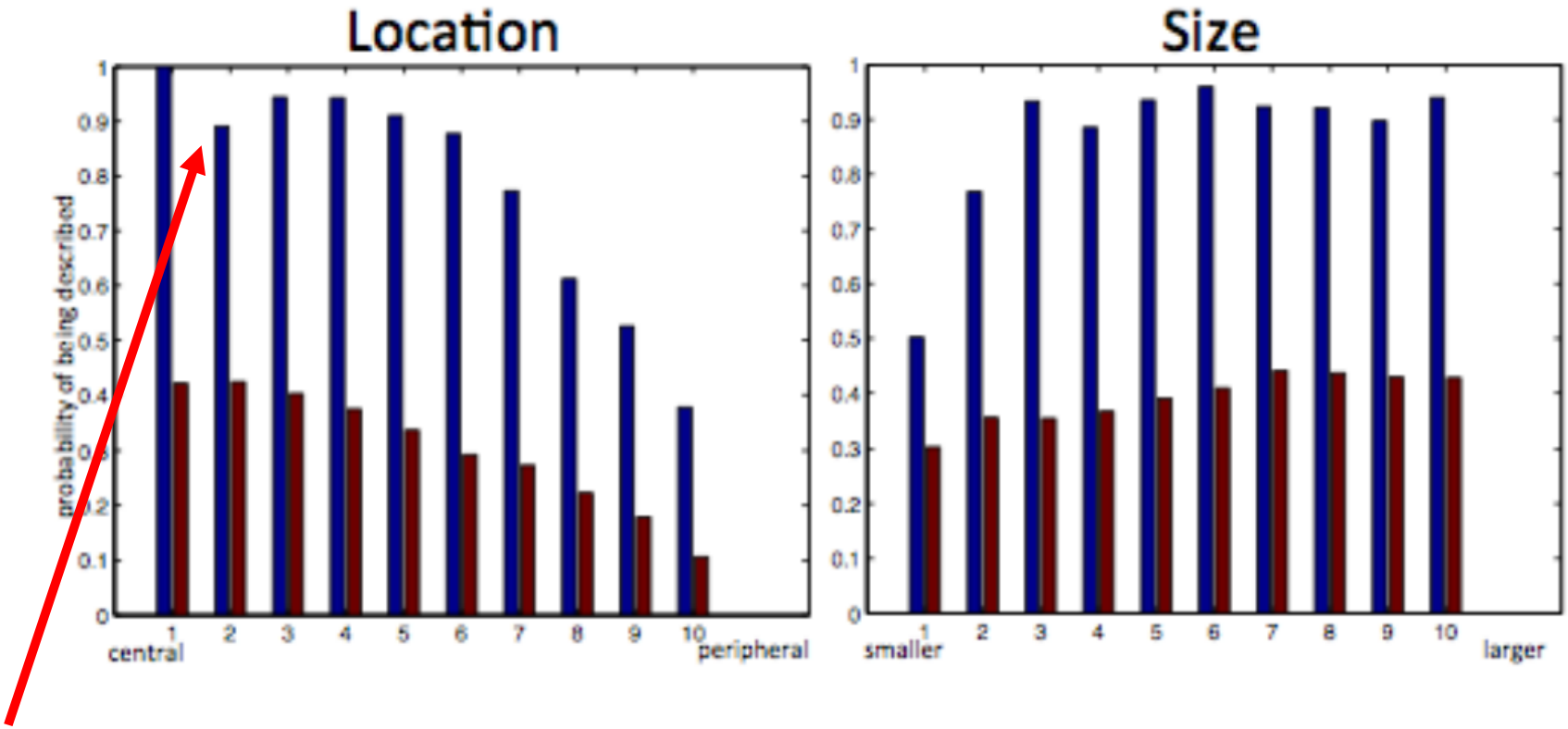
Mapping Labels to Descriptions



Result: Compositional Factors

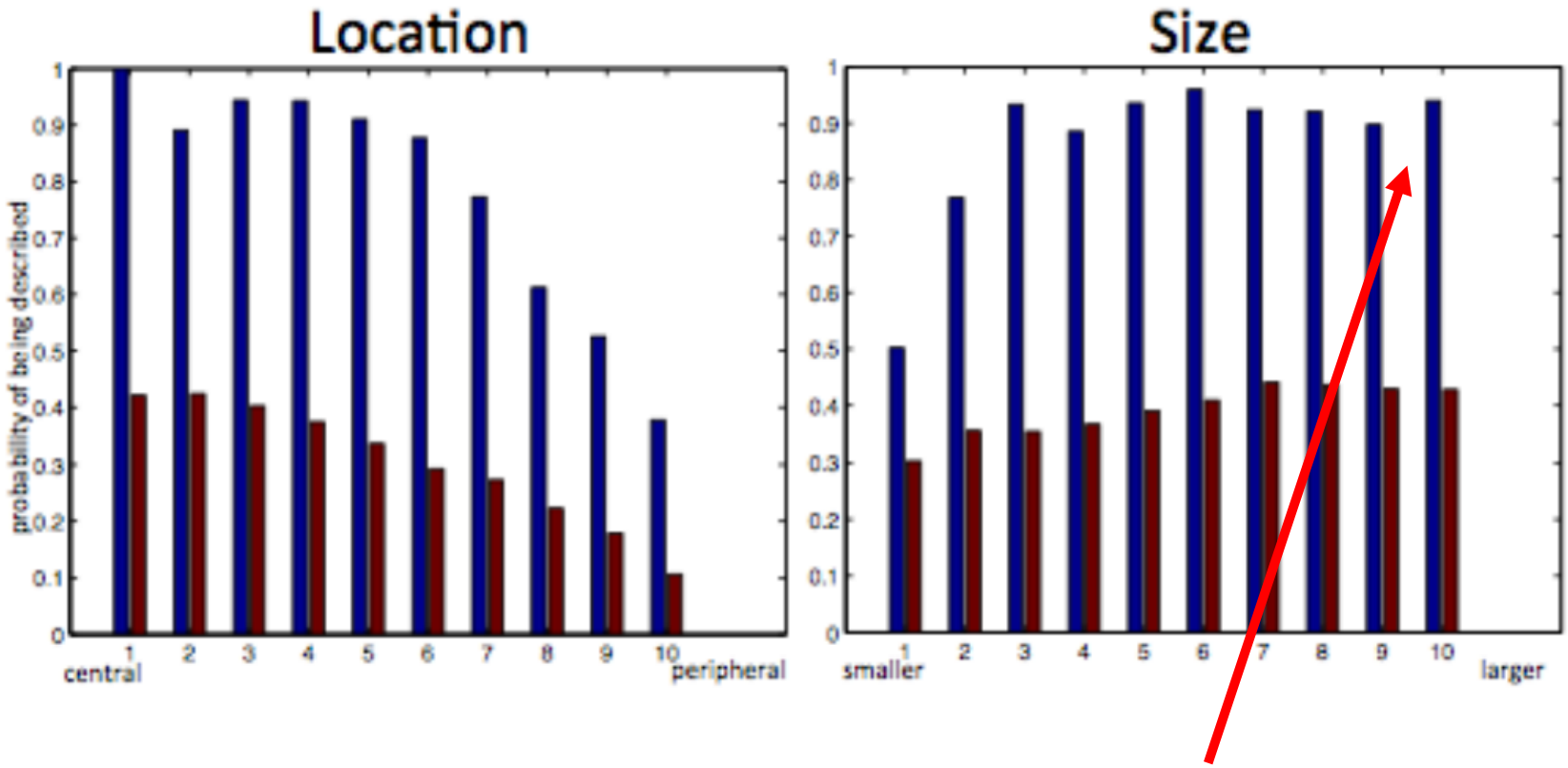


Result: Compositional Factors



Objects further from center are less likely to be mentioned

Result: Compositional Factors



Larger objects are more likely to be mentioned, unless it is too big

Semantic Factors

Top10	Prob	Last10	Prob
firework	1.00	hand	0.15
turtle	0.97	cloth	0.15
horse	0.97	paper	0.13
pool	0.94	umbrella	0.13
airplane	0.94	grass	0.13
bed	0.92	sidewalk	0.11
person	0.92	tire	0.11
whale	0.91	smoke	0.09
fountain	0.89	instrument	0.07
flag	0.88	fabric	0.07

ImageCLEF

	Prob-ImageCLEF	Prob-Pascal
Animate	0.91	0.84
Inanimate	0.53	0.55

UIUC

Semantic Factors

Top10	Prob	Last10	Prob
firework	1.00	hand	0.15
turtle	0.97	cloth	0.15
horse	0.97	paper	0.13
pool	0.94	umbrella	0.13
airplane	0.94	grass	0.13
bed	0.92	sidewalk	0.11
person	0.92	tire	0.11
whale	0.91	smoke	0.09
fountain	0.89	instrument	0.07
flag	0.88	fabric	0.07

ImageCLEF

	Prob-ImageCLEF	Prob-Pascal
Animate	0.91	0.84
Inanimate	0.53	0.55

UIUC

Some object types are more likely than others to be mentioned

Animate Objects Are more Likely to be mentioned than Inanimate

Semantic Factors – Scene Level

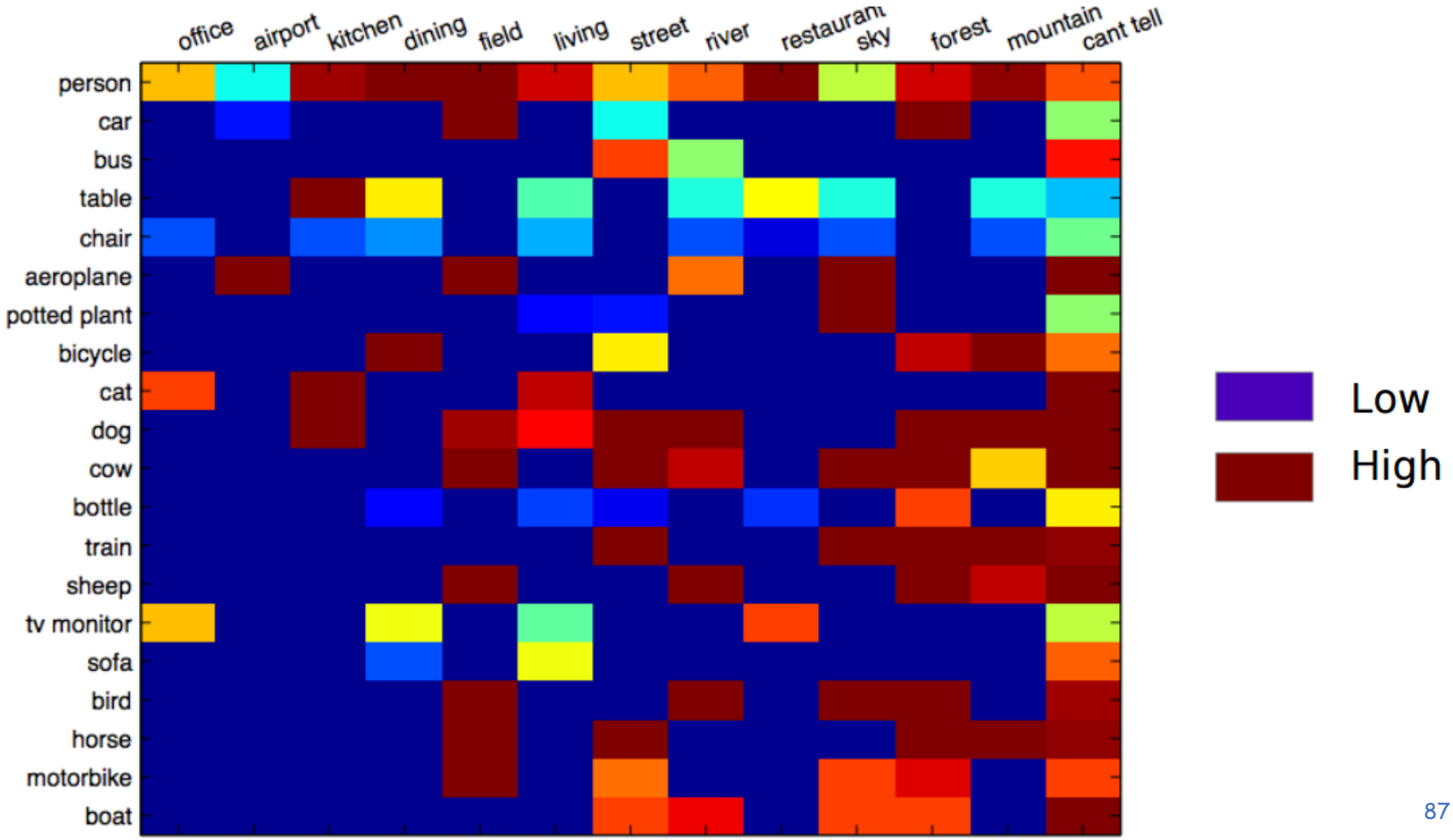
office	airport	kitchen	dining room	field	living room	street	river	restaurant	sky	forest	mountain
0.29	0.13	0.36	0.21	0.16	0.13	0.18	0.1	0.28	0.18	0.0	0.07

Probability of description for each scene type

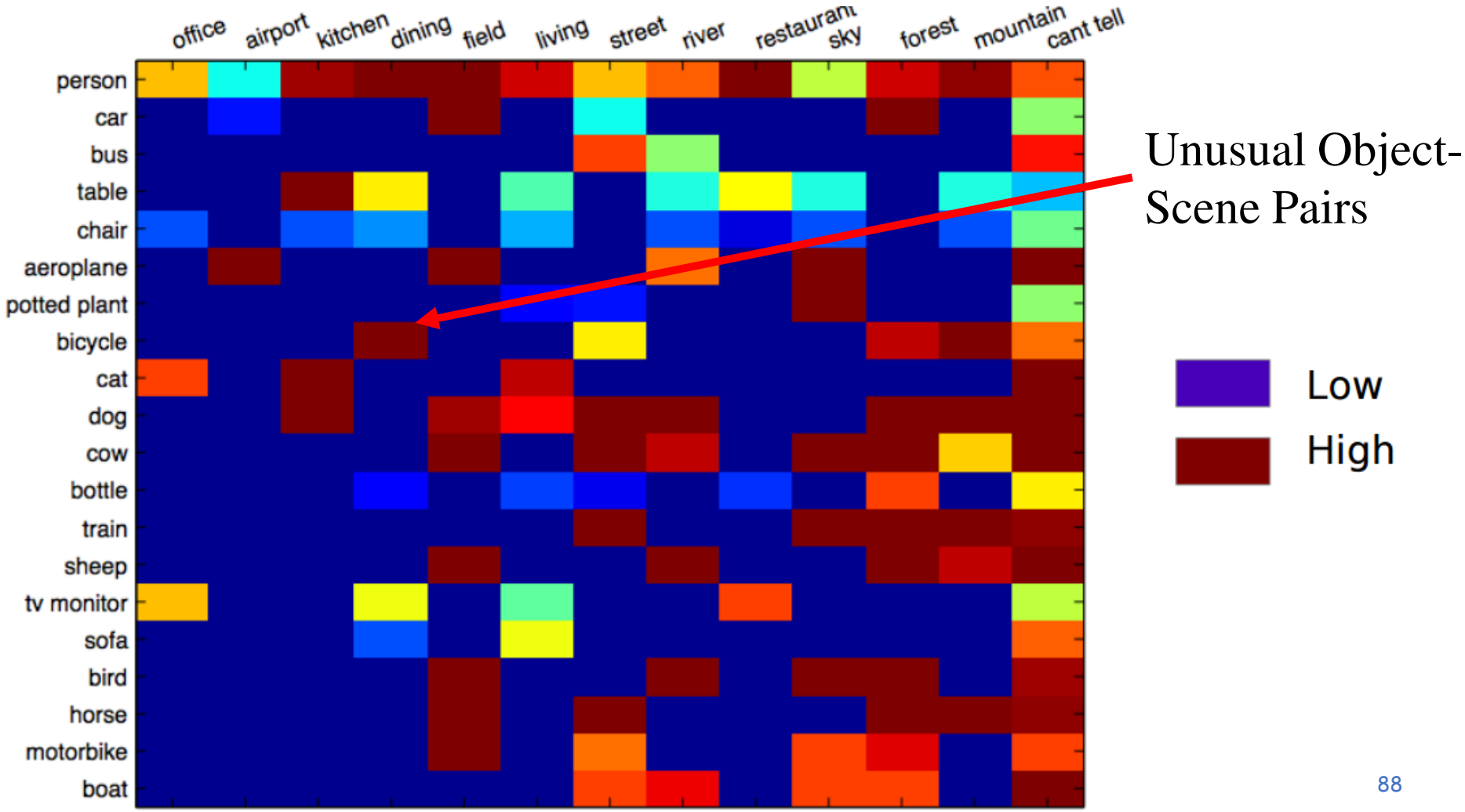
Rating	1	2	3	4	5
Prob	0.15	0.21	0.21	0.22	0.26

Scene Depiction Strength and Rating Probability

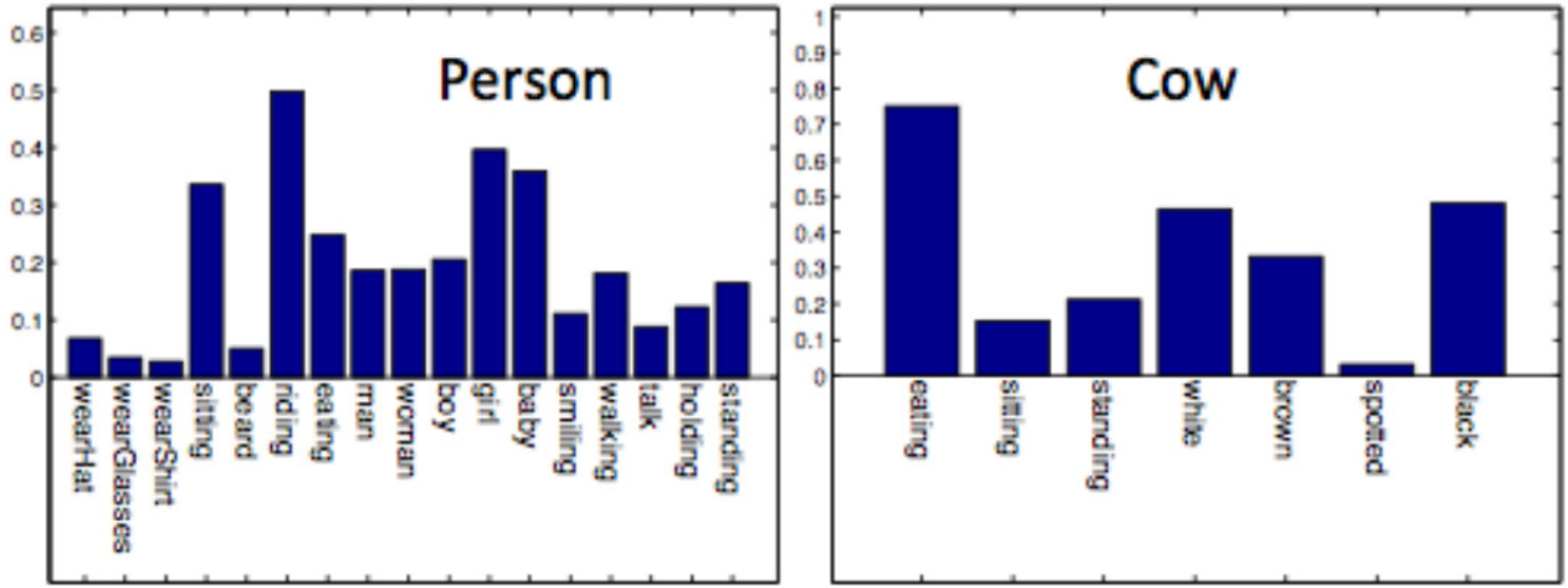
Context Factors



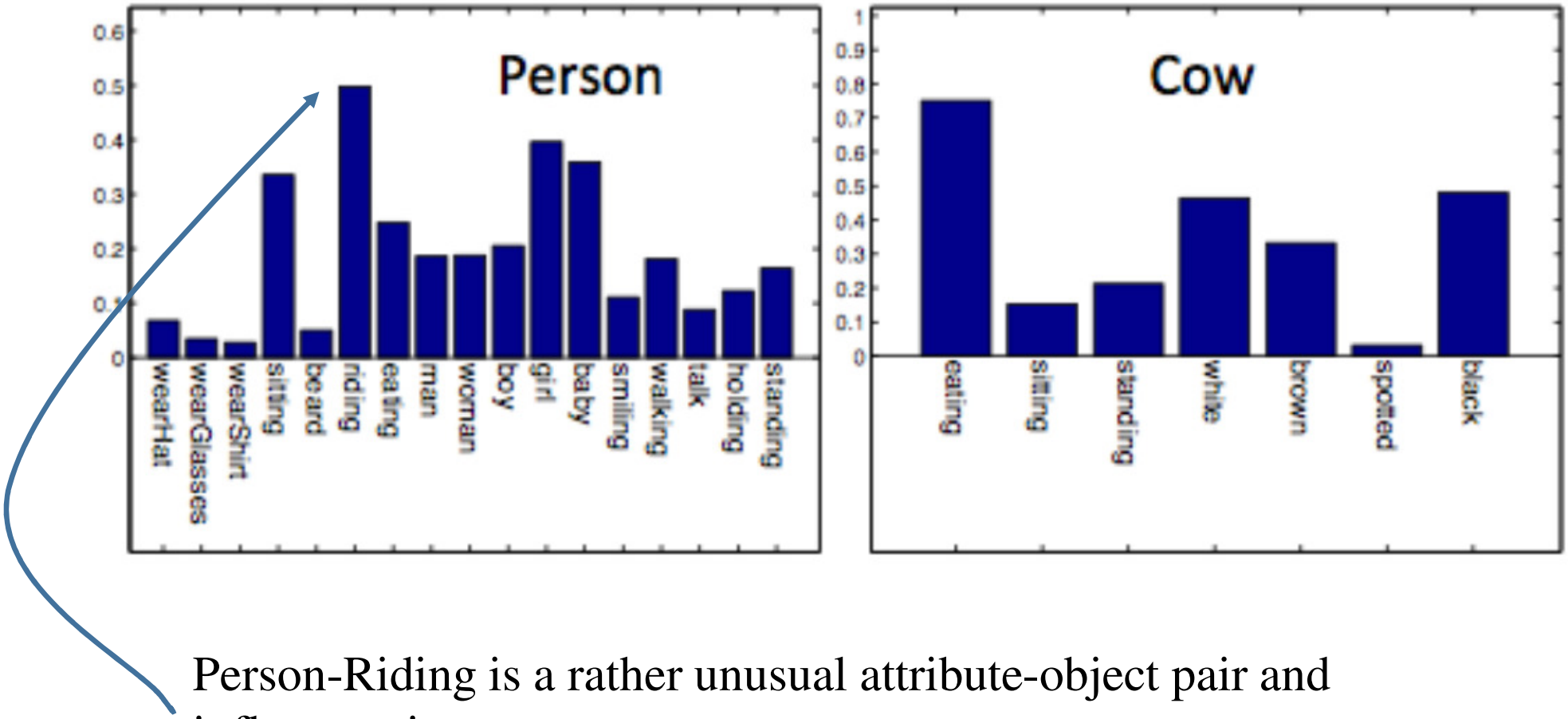
Context Factors



Attribute-Object Context

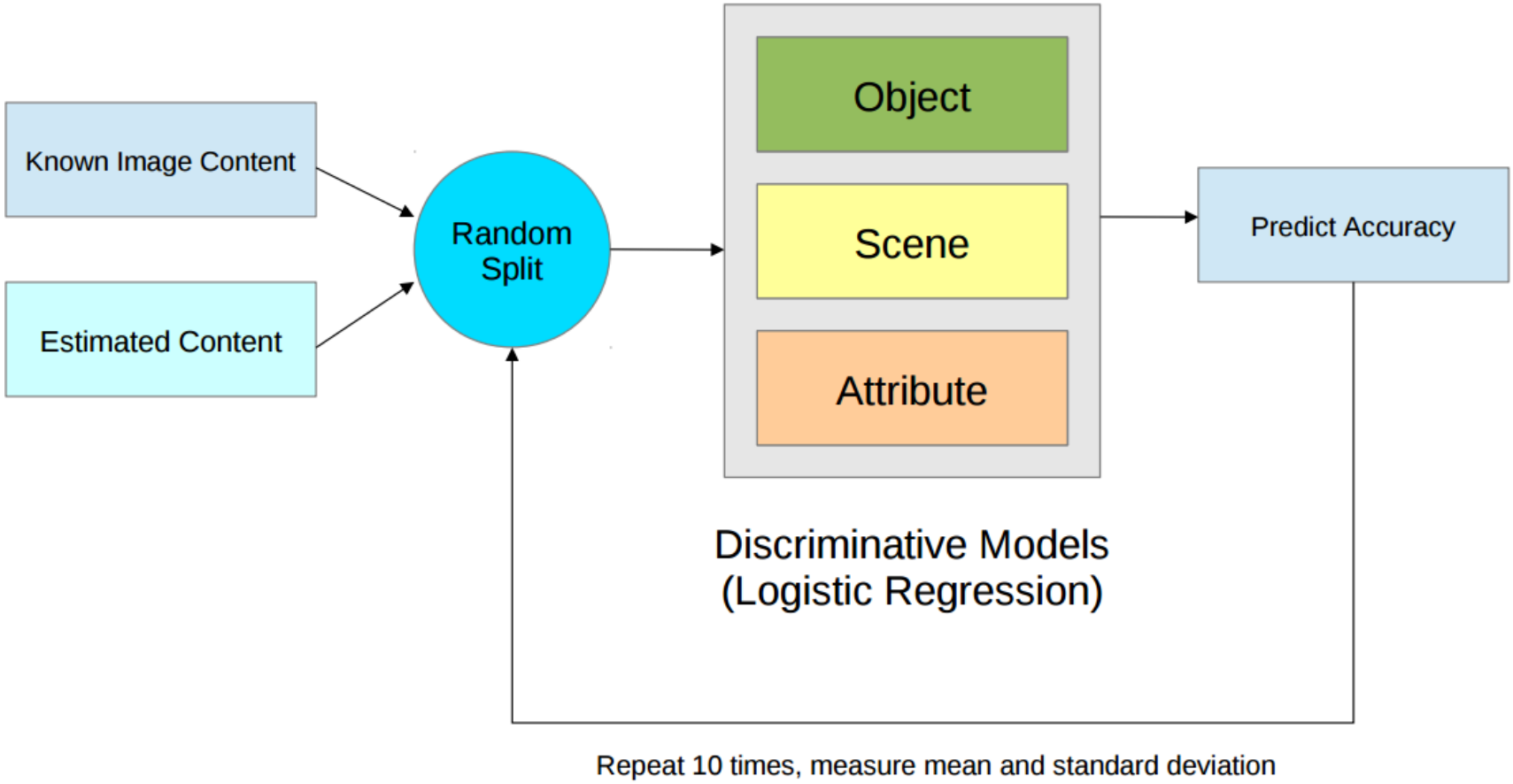


Attribute-Object Context



Person-Riding is a rather unusual attribute-object pair and influences importance

Predicting Importance



Results

Model	Features	Accuracy% (std)
Baseline (ImageCLEF)		57.5 (0.2)
Log Reg (ImageCLEF)	$K_o^s + K_o^l$	60.0 (0.1)
Log Reg (ImageCLEF)	K_o^c	68.0 (0.1)
Log Reg (ImageCLEF)	$K_o^c + K_o^s + K_o^l$	69.2 (1.4)
Baseline (UIUC-Kn)		69.7 (1.3)
Log Reg (UIUC-Kn)	$K_o^s + K_o^l$	69.9 (0.6)
Log Reg (UIUC-Kn)	K_o^c	79.8 (1.4)
Log Reg (UIUC-Kn)	$K_o^c + K_o^s + K_o^l$	82.0 (0.9)
Baseline (UIUC-Est)		76.5 (1.0)
Log Reg (UIUC-Est)	$E_o^s + E_o^l$	76.9 (1.1)
Log Reg (UIUC-Est)	E_o^c	78.9 (1.4)
Log Reg (UIUC-Est)	$E_o^c + E_o^s + E_o^l$	79.52 (1.2)

We are trying to predict whether an object will be mentioned in a description.

Baseline: Predict everything

Results

Model	Features	Accuracy% (std)
Baseline (ImageCLEF)		57.5 (0.2)
Log Reg (ImageCLEF)	$K_o^s + K_o^l$	60.0 (0.1)
Log Reg (ImageCLEF)	K_o^c	68.0 (0.1)
Log Reg (ImageCLEF)	$K_o^c + K_o^s + K_o^l$	69.2 (1.4)
Baseline (UIUC-Kn)		69.7 (1.3)
Log Reg (UIUC-Kn)	$K_o^s + K_o^l$	69.9 (0.6)
Log Reg (UIUC-Kn)	K_o^c	79.8 (1.4)
Log Reg (UIUC-Kn)	$K_o^c + K_o^s + K_o^l$	82.0 (0.9)
Baseline (UIUC-Est)		76.5 (1.0)
Log Reg (UIUC-Est)	$E_o^s + E_o^l$	76.9 (1.1)
Log Reg (UIUC-Est)	E_o^c	78.9 (1.4)
Log Reg (UIUC-Est)	$E_o^c + E_o^s + E_o^l$	79.52 (1.2)

We are trying to predict whether an object will be mentioned in a description.

K = Known

E = Estimated

Predicting Objects in Descriptions

Model	Features	Accuracy% (std)
Baseline (ImageCLEF)		57.5 (0.2)
Log Reg (ImageCLEF)	$K_o^s + K_o^l$	60.0 (0.1)
Log Reg (ImageCLEF)	K_o^c	68.0 (0.1)
Log Reg (ImageCLEF)	$K_o^c + K_o^s + K_o^l$	69.2 (1.4)
Baseline (UIUC-Kn)		69.7 (1.3)
Log Reg (UIUC-Kn)	$K_o^s + K_o^l$	69.9 (0.6)
Log Reg (UIUC-Kn)	K_o^c	79.8 (1.4)
Log Reg (UIUC-Kn)	$K_o^c + K_o^s + K_o^l$	82.0 (0.9)
Baseline (UIUC-Est)		76.5 (1.0)
Log Reg (UIUC-Est)	$E_o^s + E_o^l$	76.9 (1.1)
Log Reg (UIUC-Est)	E_o^c	78.9 (1.4)
Log Reg (UIUC-Est)	$E_o^c + E_o^s + E_o^l$	79.52 (1.2)

We are trying to predict whether an object will be mentioned in a description.

K_o^s= Known size
K_o^l= Known location
K_o^c = Known category

Predicting Scenes in Sentences

Model	Features	Accuracy% (std)
Baseline (UIUC-Kn)		86.0 (0.2)
Log Reg (UIUC-Kn)	K_s^c+K_s^r	96.6 (0.2)
Log Reg (UIUC-Est)	E_s^d	87.4 (1.3)

K_s^c = Known scene category
K_s^r = Known user rating of scene
E_s^d = Estimated scene descriptor (classification scores for 26 scene categories)

Accuracy is predicting whether or not a scene will be mentioned

Predicting Attributes in Sentences

Model	Features	Accuracy% (std)
Baseline (UIUC-Kn)		96.3 (.01)
Log Reg (UIUC-Kn)	$K_a^c + K_o^c$	97.0 (.01)
Log Reg (UIUC-Est)	$E_a^d + E_o^c$	96.7 (.01)

K_a^c = Known attribute category
K_o^c = Known object category
E_a^d = Estimated attribute descriptor
E_o^c = Estimated object detection category

Accuracy is predicting whether or not a specific attribute type will be mentioned

Conclusions

- The authors' study examines how compositional, semantic, and contextual factors predict importance in images
  - Importance is defined here as the probability of image content appearing in descriptions of the image
- Strengths:
  - The paper attempts to build predictive models of importance (not just descriptive)
  - Their evaluation breaks down the problem into known / estimated
    - This helps us eliminate poor CV performance as noise from the evaluation
  - The factors are examined and evaluated on a large scale (1k-20k) vs 97 images prior
- Weaknesses:
  - Why not consider object saliency as a metric?
  - Does task affect importance? Vs. Free viewing?
  - How about relative attributes? Could this help over binary attributes?

Discussion

References

- http://people.csail.mit.edu/tjudd/WherePeopleLook/index.html
- http://people.csail.mit.edu/tjudd/WherePeopleLook/Docs/wherepeoplelook.pdf
- http://people.csail.mit.edu/tjudd/WherePeopleLook/Docs/ICCVPoster.pdf
- http://people.csail.mit.edu/tjudd/WherePeopleLook/Docs/TeaserPoster.pdf
- http://people.csail.mit.edu/tjudd/WherePeopleLook/Docs/WherePeopleLookTalk.key
- http://people.csail.mit.edu/tjudd/WherePeopleLook/Code/JuddSaliencyModel/README.txt
- http://projects.csail.mit.edu/video/talks/defenses/tilke_judd_defense.mp4
- http://people.csail.mit.edu/tjudd/TJuddPhDThesis.pdf
- http://people.csail.mit.edu/tjudd/TJuddThesisDefenseSlides.pps_media
- http://www.scholarpedia.org/article/Visual_salience#Neural_and_computational_mechanisms
- http://www.cs.tufts.edu/~sarasu/courses/comp175-2010fa/pdf/comp175-00-saliency.pdf
- http://www.cs.utexas.edu/~cv-fall2012/slides/niveda-paper.pdf
- https://www.youtube.com/watch?v=jpIkMBsZdRY
- http://tamaraberg.com/papers/importancefactors.pdf