PinView: Implicit Feedback in Content-Based Image Retrieval

Peter Auer[1], *Zakria Hussain*[2], Samuel Kaski[3], Arto Klami[3], Jussi Kujala[3], Jorma Laaksonen[3], Alex P. Leung[1], Kitsuchart Pasupa[4], John Shawe-Taylor[2]

[1]University of Leoben, Austria

[2]University College London, UK

[3]Aalto University School of Science & Technology, Finland

[4]University of Southampton, UK

2nd September 2010

The PinView project

What is PinView?

- FP7 funded project (Jan 2008 – Jan 2011).
- Goal: "find a proactive personal information navigator that allows retrieval of multimedia - such as still images, text and video - from unannotated databases."
- A **P**ersonal **I**nformation **N**avigator Adapting Through **View**ing = **PinView**
- Implicit feedback $\rightarrow$ eye movements.
- Retrieval $\rightarrow$ content-based image retrieval.
- Website: http://www.pinview.eu/

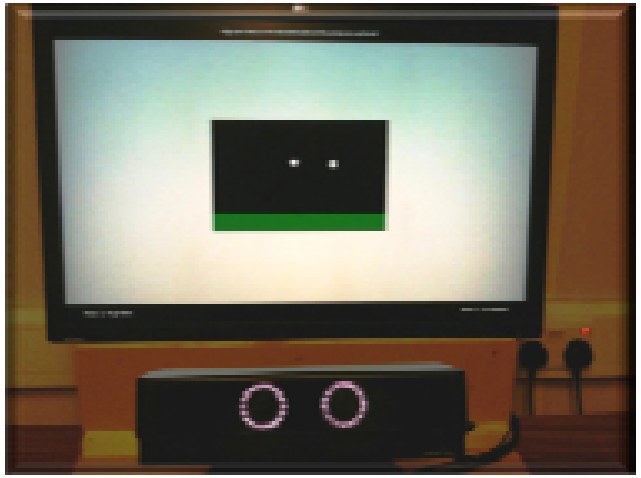
PinView

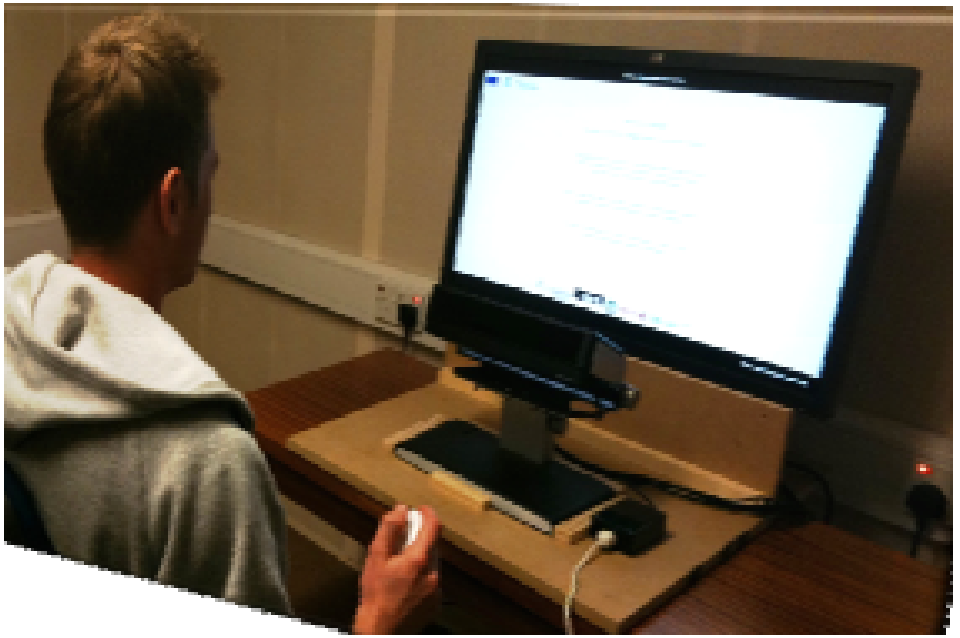
PinView consortium

Who's involved?

- Aalto University School of Science & Technology, Finland.
- University College London, UK.
- University of Southampton, UK.
- University of Leoben, Austria.
- Xerox research, France. (Industrial)
- Celum gmbh, Austria. (Industrial)

Some motivation





PinView

Content-Based Image Retrieval with Relevance-Feedback

- CBIR: Find relevant images in an unannotated database.
- "Content-Based" $\rightarrow$ analyse the actual contents of images, not keywords, tags, or image descriptors.
- "Relevance-Feedback" $\rightarrow$ user provides feedback on relevance of retrieved images during a search.
  - Explicit feedback: user states relevance of images by some direct method, i.e., pointer clicks.
  - **Implicit feedback**: predict relevance of images based on users behaviour, i.e., eye movements.

PinView

PinView system components

- Relevance prediction $\rightarrow$ predict relevance of images based on implicit feedback.
- Exploration-Exploitation $\rightarrow$ explore new images, and exploit close-by images to those considered relevant.
- Learning the metric $\rightarrow$ generate a richer metric space for EE using several different types of features extracted from images.
- Feature selection $\rightarrow$ incorporate the eye movement features together with the learnt metric.

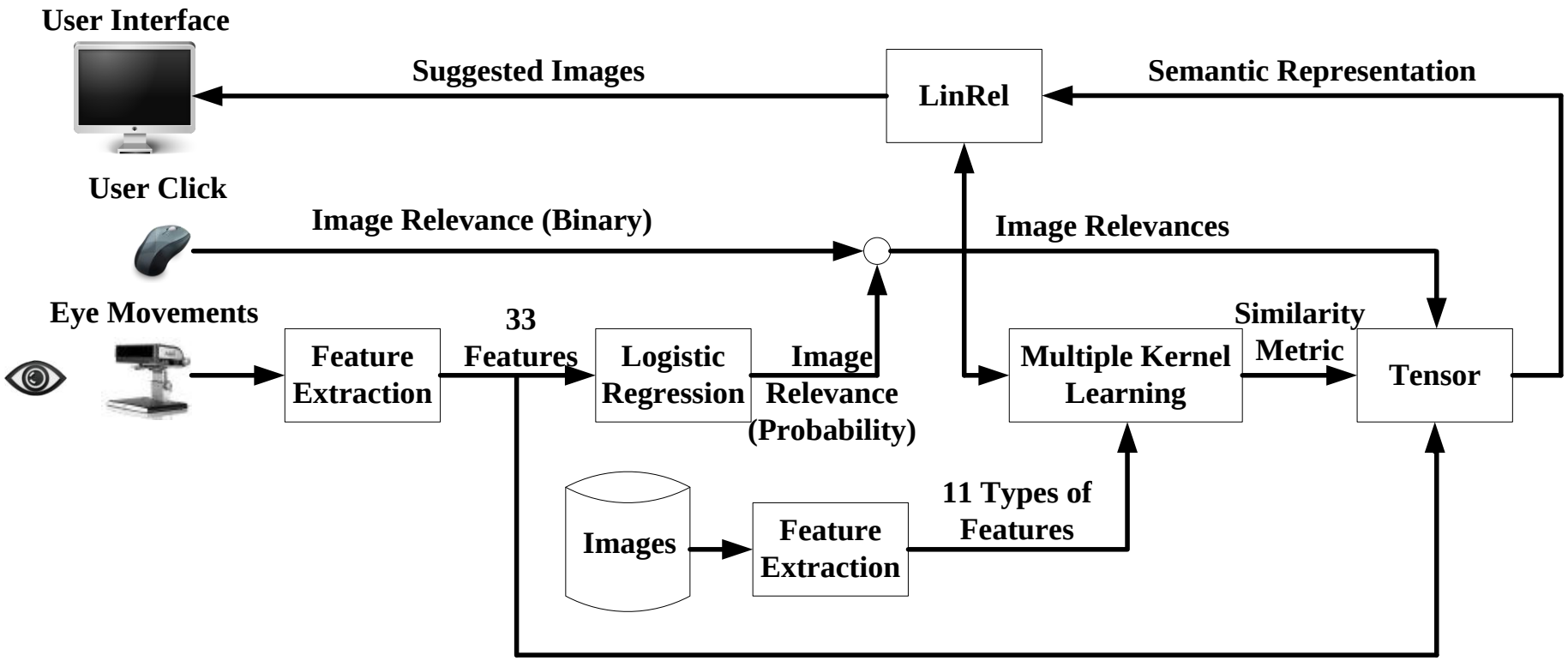
PinView

Component algorithms

- Logistic Regression.
- Associative Reinforcement Learning with Linear Value Functions (LinRel).
- Multiple Kernel Learning (MKL).
- Tensor SVM.

PinView

PinView


User Interface
Suggested Images
LinRel
Semantic Representation
User Click
Image Relevance (Binary)
Image Relevances
Eye Movements
Feature Extraction
33 Features
Logistic Regression
Image Relevance (Probability)
Multiple Kernel Learning
Similarity Metric
Tensor
Images
Feature Extraction
11 Types of Features

Eye movement features for relevance prediction

Table: Eye movement features collected in PinView.

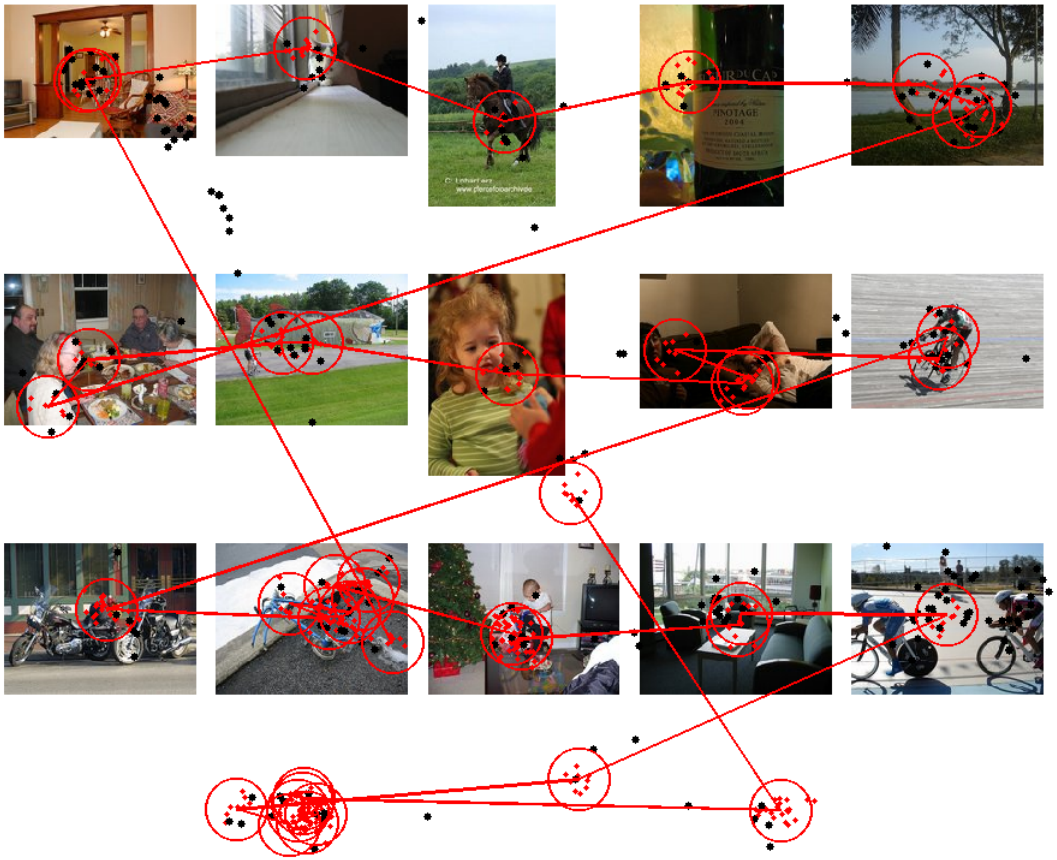
Number	Name	Description
Raw data features		
1	numMeasurements	log of total time of viewing the image
2	numOutsideFix	total time for measurements outside fixations
3	ratioInsideOutside	percentage of measurements inside/outside fixations
4	speed	average distance between two consecutive measurements
5	coverage	number of subimages covered by measurements[1]
6	normCoverage	coverage normalized by numMeasurements
7	pupil	maximal pupil diameter during viewing
8	nJumps1	number of breaks longer than 60ms[2]
9	nJumps2	number of breaks longer than 600ms[2]
Fixation features		
10	numFix	total number of fixations
11	meanFixLen	mean length of fixations
12	totalFixLen	total length of fixations
13	fixPrct	percentage of time spent in fixations
14	nJumpsFix	number of re-visits to the image
15	maxAngle	maximal angle between two consecutive saccades[3]
16	firstFixLen	length of the first fixation
17	firstFixNum	number of fixations during the first visit
18	distPrev	distance to the fixation before the first
19	durPrev	duration of the fixation before the first

[1] Image divided into a grid of 4x4 subimages – covering a subimage means that at least one measurement falls within it.
[2] A sequence of measurements outside the image occurring between two consecutive measurements within the image.
[3] A transition from one fixation to another.

PinView

Example collage

Figure: Red circles mark fixations and small red dots correspond to raw measurements that belong to those fixations. The black dots mark raw measurements that were not included in any of the fixations.



PinView

Relevance Prediction

- Predict relevance of images based on eye movement data.
- We extract 33 statistical features, i.e., total time image was looked at, fixation time, etc.
- Train logistic regression model on data set of previously collected online search sessions.
- Use the logistic model to predict relevance of an image based on these 33 eye movement features.

PinView

Exploration-Exploitation

- Find images close to those indicated as relevant (Exploitation).
- Find images further away (Exploration).
- Use the LinRel algorithm.

PinView

LinRel algorithm for selecting images to present

- Objective: maximize the number of relevant images presented to the user, $\sum_t y_t$, where y_t is the relevance of the t-th presented image.
- Assumption: The expected value of the relevance y_I is a linear function of the image features $\mathbf{x}_I$,

$$E[y_I] = \mathbf{w}^\top \mathbf{x}_I.$$

PinView

Exploration with LinRel

- In iteration t, LinRel estimates $\mathbf{w}$ from previously presented images $\mathbf{X}_t = (\mathbf{x}_1, \ldots, \mathbf{x}_{t-1})^\top$ and the observed relevance scores $\mathbf{y}_t = (y_1, \ldots, y_{t-1})^\top$, using linear regression:

$$\hat{\mathbf{w}}_t = \arg\min_{\mathbf{w}} \|\mathbf{X}_t\mathbf{w} - \mathbf{y}_t\|_2^2$$

- The estimated weight vector $\hat{\mathbf{w}}_t$ predicts the relevance of image I as $\hat{y}_I = \hat{\mathbf{w}}_t^\top \mathbf{x}_I$.
- Predictions might be misleading if $\hat{\mathbf{w}}_t$ is inaccurate.
- LinRel optimistically selects images which *might* be most relevant, given the variance σ_I^2 of $\mathbf{y}_I$.
- The image I with maximum $\hat{\mathbf{w}}_t^\top \mathbf{x}_I + c\sigma_I$ is selected.

PinView

LinRel

- In each iteration t the regularised LinRel algorithm calculates

$$\mathbf{a}_I = \mathbf{x}_I \cdot (\mathbf{X}_t^\top \mathbf{X}_t + \mu \mathbf{I})^{-1} \mathbf{X}_t^\top, \tag{1}$$

  for each image I.
- Select for presentation the images which maximise

$$\mathbf{a}_I \cdot \mathbf{y}_t + \frac{c}{2} \|\mathbf{a}_I\|_2, \tag{2}$$

  for some specified constant $c > 0$.

PinView

Kernel LinRel

- To kernelize LinRel

$$\mathbf{a}_I = \big(k(I, I_1) \ \cdots \ k(I, I_{t-1})\big) \cdot (\mathbf{K}_t + \mu \mathbf{I})^{-1},$$

where $I_1, \ldots, I_{t-1}$ are the images selected in iterations $i = 1, \ldots, t-1$ and $\mathbf{K}_t$ is the Gram matrix

$$\mathbf{K}_t = (k(I_i, I_j))_{1 \le i,j \le t-1}.$$

Thus $\mathbf{a}_I$ can be calculated by using only the kernel function $k(\cdot, \cdot)$.

- Since the selection rule (2) remains unchanged, this gives the kernelized version of LinRel.

PinView

Open problem: How to select collages of n images at once?

Some possibilities:

1. Select the n images with maximal upper confidence bounds $\hat{\mathbf{w}}_t^\top \mathbf{x} + c\sigma$.
2. Select one image with maximal upper confidence bound, and select $n-1$ images with maximal relevance estimates $\hat{\mathbf{w}}_t^\top \mathbf{x}$.
3. Do something more complicated: Iteratively select n images by their upper confidence bounds, but modify $\hat{\mathbf{w}}$ by adding the already selected images and their estimated relevance $\hat{\mathbf{w}}^\top \mathbf{x}$. (This shrinks the confidence intervals for similar images.)

PinView

Learning the metric

- We have 11 different feature extraction methods, including SIFT, Colour, Haar transform, etc.
- Each feature extraction method $\rightarrow$ a kernel.
- We would like to learn a kernel using multiple kernel learning:

$$k_{\boldsymbol{\eta}}(I, J) = \sum_{i=1}^{N} \eta_i \, k_i(I, J),$$

  where the $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_N)$ are the weights of the kernel functions $k_i(I, J)$ between images I and J.
- After each collage is shown we can learn a new metric (run MKL), and pass this new kernel to the kernel LinRel algorithm.

PinView

Multiple Kernel Learning

- Let $\mathbf{w}^k$ denote the weight vector of the kth feature space, then the MKL problem we solve is:

$$\min_{\mathbf{w}^k, \boldsymbol{\xi}} \quad \lambda \underbrace{\left(\sum_{k=1}^{K} \|\mathbf{w}^k\|_2 \right)^2}_{1-\text{norm}} + (1-\lambda) \underbrace{\sum_{k=1}^{K} \|\mathbf{w}^k\|_2^2}_{2-\text{norm}} + C\|\boldsymbol{\xi}\|_1,$$

  where $\lambda \in [0, 1]$, C is the penalty parameter and $\boldsymbol{\xi}$ denotes the slack variables.

- Regularisation: $\lambda \leftarrow 1$ is 1-norm regularisation, and $\lambda \leftarrow 0$ is 2-norm regularisation.

PinView

Feature selection using eye movements

- Would like to improve results by incorporating eye movement features together with image features.
- However, eye movement features for unseen images not known.
- We can use the technique of a Tensor SVM, where we decompose the joint feature vector found for the kernel:

$$\mathbf{K}_{\text{img}} \circ \mathbf{K}_{\text{eye}}$$

- Decomposition: SVD like procedure, to decompose joint weight vector into its component parts.
- Compute a new kernel matrix (for all image features) and pass this to the kernel LinRel algorithm.

PinView

Experimental setup

- Subset of PASCAL Visual Object Classes Challenge 2007 (VOC2007) dataset. No. of images = 2501.
- perform offline experiments with simulated search sessions.
- Search session: Iteratively use PinView to select a total of 10 collages with 15 images in each.
- The target of each search session is one of the categories.
- Total number of the search sessions in each category is 40.
- Quality measure: Precision-Recall (i.e., Average Precision).
- Eye movements required were recorded with Tobii 1750 eye movement tracker in separate online experiments.

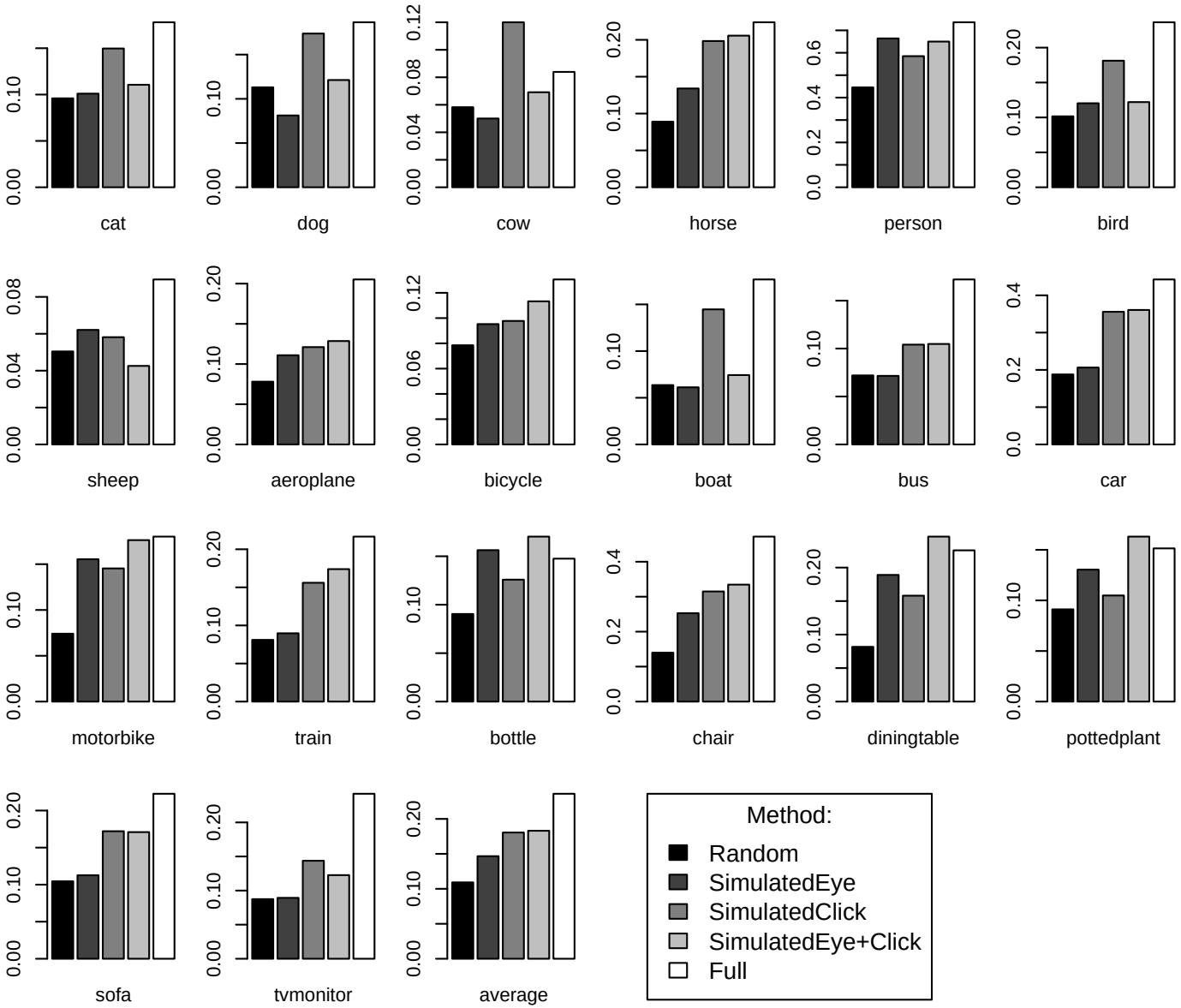
PinView

Feedback modalities

- FULL: gives true label of seen images.
- SIMULATEDCLICK: a random relevant image on collage is selected as clicked.
- SIMULATEDEYE: simulate eye movements from previous online experiments to predict relevance of images.
  - keep pool of relevant and nonrelevant image eye movement features, and then sample from these two groups in the offline experiments.
- SIMULATEDEYE+CLICK: generates the relevance values from both simulated eye movements and clicks.

PinView

Results



PinView

Conclusions

- Proposed a new CBIR system that uses relevance feedback from eye movements (PinView).
- Can infer relevance of images relatively well from eye movements.
- We can unobtrusively improve user experience by adapting user interface to the interests of the user.
- Incorporated several different machine learning algorithms into the PicSOM system – allowing the use of real world image databases.
- In the future we plan to perform online experiments on real subjects.

PinView