JMLR: Workshop and Conference Proceedings 11 (2010) 51–57 Workshop on Applications of Pattern Analysis

Pinview: Implicit Feedback in Content-Based Image Retrieval

Peter Auer[*] AUER@UNILEOBEN.AC.AT
University of Leoben, Austria

Zakria Hussain Z.HUSSAIN@CS.UCL.AC.UK
University College London, United Kingdom

Samuel Kaski SAMUEL.KASKI@TKK.FI
Arto Klami ARTO.KLAMI@TKK.FI
Jussi Kujala JUSSI.KUJALA@IKI.FI
Jorma Laaksonen JORMA.LAAKSONEN@TKK.FI
Aalto University School of Science and Technology, Finland

Alex P. Leung ALEX.LEUNG@UNILEOBEN.AC.AT
University of Leoben, Austria

Kitsuchart Pasupa K.PASUPA@GMAIL.COM
University of Southampton, United Kingdom

John Shawe-Taylor JST@CS.UCL.AC.UK
University College London, United Kingdom

Editors: Tom Diethe, Nello Cristianini, John Shawe-Taylor

Abstract

This paper describes Pinview, a content-based image retrieval system that exploits implicit relevance feedback during a search session. Pinview contains several novel methods that infer the intent of the user. From relevance feedback, such as eye movements or clicks, and visual features of images Pinview learns a similarity metric between images which depends on the current interests of the user. It then retrieves images with a specialized reinforcement learning algorithm that balances the tradeoff between exploring new images and exploiting the already inferred interests of the user. In practise, we have integrated Pinview to the content-based image retrieval system PicSOM, in order to apply it to real-world image databases. Preliminary experiments show that eye movements provide a rich input modality from which it is possible to learn the interests of the user.

1. Introduction

The need to find interesting images from a large collection is common to, for instance, laymen surfing the internet or professional graphic designers in their work. In content-based image retrieval (CBIR) the system tries to show the user images that are visually similar to the ones it thinks the user is looking for. Unfortunately, we typically do not have a very clear idea what the user is looking for. A common approach is to ask for explicit

[*] The authors appear in alphabetical order.

relevance feedback on the images shown to the user to get a rough idea of the search target. However, this is laborous to the user.

Another approach is to obtain the relevance feedback implicitly, by measuring attention patterns of the users and inferring the relevance of the seen images from these (Kelly and Teevan, 2003; Klami et al., 2008). This is the approach taken by Pinview, a CBIR system presented in this paper. It uses eye movements as implicit relevance feedback on the images to infer the relevance of seen images in order to proactively show new images. Pinview must solve several subproblems to take advantage of the recorded implicit feedback. Which features of the images are interesting and how does the user perceive them? Given that we are able to show the user only a limited number of images, how should we balance the fact that we should both exploit our limited knowledge of the query and explore new kinds of images? Section 2 presents details on how Pinview answers these questions. Section 3 gives results of preliminary experiments we performed with the Pinview system. Finally, Section 4 concludes.

2. System Components

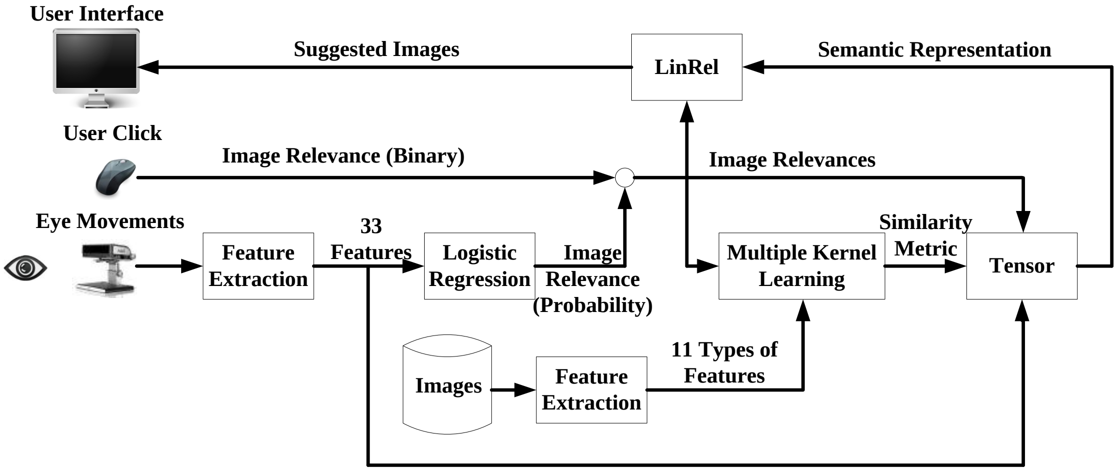

Figure 1: Main components and data flow in Pinview.

This section describes the main components of the PinView CBIR system, visually summarized in Figure 1. From eye movements the system predicts relevances of seen images. Tensor decomposition and multiple kernel learning algorithms then infer a metric between images using known low-level visual features of the images and relevance feedback on the seen images. Finally, specialized exploration-exploitation algorithm LinRel suggests new images to be shown to the user, and they are referred from a database and displayed through the PicSOM backend (Laaksonen et al., 2002). In the following sections we will go through these components in more detail.

2.1. Relevance Prediction

Pinview infers relevance of images during a search task from eye movements of the user. For each seen image Pinview extracts 33 statistical features including the logarithm of the total time the image was looked at and features capturing regressions to already seen images. The relevance of an image is predicted from the features using a logistic regression model trained on a data set of previously collected online search sessions. Similar relevance prediction has been applied earlier in (Puolamäki et al., 2005).

2.2. Tensor Decomposition

There exists some kind of a relationship between eye movements and image features. We learn the relationship by using tensor representation which creates an implicit correlation space. The tensor representation can simply be computed by taking dot products between each individual kernel matrix of each view (Szedmak et al., 2005; Pulmannová, 2004). Then we use this kernel to train a tensor kernel SVM (Hardoon and Shawe-Taylor, 2010) to generate a weight matrix which is composed of both views. As we do not have the eye movement features for images not yet displayed to the user, we need to decompose the weight matrix into one weight vector per view. This has been resolved by (Hardoon and Shawe-Taylor, 2010) who propose a novel singular value decomposition (SVD) like approach for decomposing the resulting tensor weight matrix into its two component parts, without needing to directly access the feature space.

A preliminary study of combining eye movement and image features using tensor decomposition with the Ranking SVM can be found in (Hardoon and Pasupa, 2010).

2.3. Multiple Kernel Learning

Learning the similarity measures or metric of importance for our CBIR task is of upmost importance. Some image searches may require a combination of image features to quickly distinguish the relevant from other less relevant images. For instance, colour and texture features may be important to find pictures of snowscapes, whereas colour may be the only important feature needed to find images of blue skies. We would like to use a combination of the metrics as a cue to finding relevant images quickly and efficiently, and then pass this learnt metric (kernel) to the LinRel algorithm to be described in Section 2.4.

Multiple kernel learning (MKL) attempts to find a combination of kernels by solving a classification problem using a weighted combination of kernels. Given that our Pinview system will use several different feature extraction methods provided by the PicSOM engine, we view each one as a separate feature space – hence, giving us N different kernels $\mathcal{K} = \{k_1, \ldots, k_N\}$. Using MKL we construct the kernel function

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

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. We follow a simpler formulation of the algorithm described in (Hussain et al., 2008), which uses a parameter $\lambda \in [0, 1]$ in order to move between a 1-norm regularisation (when

$\lambda = 1$) and a 2-norm regularisation (when $\lambda = 0$). The justification of using this algorithm is that we expect to use many kernels in the beginning of the search and not too many near the end, as during the session we gain a better understanding of relevance inferred through (explicit) pointer clicks and (implicit) eye movements (as described in Section 2.1).

2.4. LinRel

In CBIR the search engine faces a trade-off between exploration and exploitation. It can maintain an implicit or explicit representation of the estimate $\hat{\mathbf{w}}$ of the unknown weight vector $\mathbf{w}$ which maps image features to relevance scores. When selecting the next image for presentation to the user, the search engine might simply select the image with the highest estimated relevance score based on $\hat{\mathbf{w}}$. But since the estimate $\hat{\mathbf{w}}$ might be inaccurate, this exploitative choice might be suboptimal. Alternatively, the search engine might exploratively select an image for which the user feedback improves the accuracy of the estimate $\hat{\mathbf{w}}$, enabling better image selections in subsequent iterations.

In each iteration t, the LinRel algorithm obtains an estimate $\hat{\mathbf{w}}_t$ by solving the linear regression problem (Auer, 2002) $\mathbf{y}_t \approx \mathbf{X}_t \cdot \hat{\mathbf{w}}_t$, where $\mathbf{y}_t = (y_1 \ \cdots \ y_{t-1})^\top$ is the column vector of relevance scores received so far, and $\mathbf{X}_t = (\mathbf{x}_1 \ \cdots \ \mathbf{x}_{t-1})^\top$ is the matrix of row feature vectors of the images presented so far. Based on the estimated weight vector $\hat{\mathbf{w}}$, LinRel calculates an estimated relevance score $\hat{y}_I = \mathbf{x}_I \cdot \hat{\mathbf{w}}$ for each image I that has not already been presented to the user. To deal with the exploration-exploitation trade-off, LinRel selects for presentation not the image with largest estimated relevance score, but the image with the largest upper confidence bound for the relevance score. The upper confidence bound for an image I is calculated as $\hat{y}_I + c\hat{\sigma}_I$, where $\hat{\sigma}_I$ is an upper bound on the standard deviation of the relevance estimate $\hat{y}_I$. The constant c is used to adjust the confidence level of the upper confidence bound.

In each iteration t the regularized LinRel algorithm for $n = 1$ (Auer et al., 2009), 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 and selects for presentation the images which maximize

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

for some specified constant $c > 0$.

Kernelization. Kernel learning can be integrated into the LinRel algorithm. A kernel learning algorithm learns a suitable metric between images in respect to a user query, by finding a good kernel function. Since in each iteration the kernelized LinRel algorithm relies on a fixed kernel function, the integration of kernel learning into LinRel is very simple: LinRel calls the kernel learning algorithm in the beginning of each iteration and then uses the kernel matrix returned by the kernel learning algorithm. To kernelize LinRel (Auer et al., 2009),

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

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

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

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.

3. Experiments

We have carried out a set of preliminary experiments on the Pinview system.

Data sets. We use a subset of the PASCAL Visual Object Classes Challenge 2007 (VOC2007) dataset. The number of images in the data set is 2501. It contains 20 non-exclusive image categories shown in Table 1.

Setup. We perform offline experiments with simulated search sessions. In each search session we 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. We record the performance of Pinview with three possible feedback modalities which are described in the next paragraph. We also record the performance of a random baseline which simply returns random unseen images. The total number of the search sessions in each category is 30. The quality measure is precision that is the number of relevant images retrieved divided by the total number of retrieved images. The eye movements required by simulation were recorded with Tobii 1750 eye movement tracker in separate online experiments.

Feedback modalities. We use three different feedback modalities in the experiments. The first feedback modality Full gives the true label of each seen image to Pinview. The second feedback modality Noisy is a noisy version of Full, where each nonrelevant label is independently flipped to relevant with probability 0.346 and each relevant label is flipped to nonrelevant with probability 0.244 (these numbers were estimated from the noise of eye movement relevance prediction in online experiments). The final feedback modality Simulated gives simulated eye movements to Pinview. The simulated eye movements are selected from a pool of previously recorded eye movements from online experiments. We split the eye movements to a positive group and a negative group depending on whether the image was relevant or nonrelevant in the task where it was recorded. In the offline experiment we sample eye movements from the positive group for relevant images and from the negative group for nonrelevant images. Hence, we are able to approximate the online performance of Pinview in an offline experiment.

Table 1 gives the results. Full feedback corresponds to the performance of Pinview under ideal conditions, where the user is able and willing to provide perfect feedback. Noisy and Simulated provide lower bounds for the performance using only the implicit feedback. The real mean performance of the full system in online experiments is between these two numbers. Note that the results are noisy which is clearly visible in the results of the individual categories, even to the extent that the explicit feedback is harmful in four categories. Also, in these experiments the average precision was maximized, which might have had a negative impact on the performance of individual categories.

Table 1: Precision of Pinview in percentage points with different input modalities compared to the random baseline on several categories.

Category	Baseline	Full	Noisy	Simulated
Cat	7.7	26.6	7.4	5.5
Dog	9.7	9.9	9.7	7.9
Cow	3.1	1.6	3.6	4.0
Horse	7.2	9.5	10.8	10.5
Person	44.0	71.6	65.0	68.0
Bird	7.4	22.8	8.9	4.5
Sheep	14.1	26.4	14.0	10.6
Aeroplane	8.9	48.9	18.9	25.0
Bicycle	4.7	4.2	6.7	8.0
Boat	3.3	26.4	2.9	1.0
Bus	4.6	9.9	4.4	2.1
Car	16.2	36.9	5.9	16.5
Motorbike	4.8	2.4	6.6	9.0
Train	4.4	14.1	6.2	4.5
Bottle	5.9	10.9	8.3	12.0
Chair	25.8	46.4	36.3	33.8
Diningtable	12.9	19.2	18.8	21.5
Pottedplant	11.1	14.3	13.2	19.8
Sofa	18.2	18.1	20.4	23.1
Tv-monitor	8.2	28.1	11.5	15.1
Average	11.1	22.4	15.1	15.0

4. Conclusions

This paper described the first version of CBIR system called Pinview which records implicit relevance feedback signals from the users and infers the intent of the user using several novel machine learning methods. Our results indicate that we can infer relevance of images relatively well from eye movements. Hence, we can unobtrusively improve user experience by adapting user interface to the interests of the user. In the future we plan to perform online experiments on real subjects.

Acknowledgments. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007–2013) under *grant agreement* n° 216529, Personal Information Navigator Adapting Through Viewing, PinView, and IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. This publication only reflects the authors' views.

References

Peter Auer. Using confidence bounds for exploration-exploitation trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2002.

Peter Auer, Alex Leung, Zakria Hussain, and John Shawe-Taylor. Report on using side information for exploration-exploitation trade-offs. PinView FP7-216529 Deliverable D4.2.1, 2009. URL `http://www.pinview.eu/`.

David R. Hardoon and Kitsuchart Pasupa. Image ranking with implicit feedback from eye movements. In *Proceedings of ETRA 2010: ACM Symposium on Eye-Tracking Research & Applications*, pages 291–298. ACM, 2010.

David R. Hardoon and John Shawe-Taylor. Decomposing the tensor kernel support vector machine for neuroscience data with structure labels. *Machine Learning Journal: Special Issue on Learning From Multiple Sources*, 79(1-2):29–46, 2010. ISSN 0885-6125.

Zakria Hussain, Kitsuchart Pasupa, Craig J. Saunders, and John Shawe-Taylor. Basic metric learning. PinView FP7-216529 Deliverable D3.1, 2008. URL `http://www.pinview.eu/`.

Diane Kelly and Jaime Teevan. Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum*, 37(2):18–28, 2003.

Arto Klami, Craig Saunders, Teófilo E. de Campos, and Samuel Kaski. Can relevance of images be inferred from eye movements? In *MIR '08: Proceeding of the 1st ACM International Conference on Multimedia Information Retrieval*, pages 134–140, New York, NY, USA, 2008. ACM.

Jorma Laaksonen, Markus Koskela, and Erkki Oja. PicSOM — self-organizing image retrieval with MPEG-7 content descriptions. In *Networks, Special Issue on Intelligent Multimedia Processing*, pages 841–853, 2002.

Sylvia Pulmannová. Tensor products of hilbert space effect algebras. *Reports on Mathematical Physics*, 53(2):301–316, 2004.

Kai Puolamäki, Jarkko Salojärvi, Eerika Savia, Jaana Simola, and Samuel Kaski. Combining eye movements and collaborative filtering for proactive information retrieval. In *Proceedings of SIGIR 2005, Twenty-Eighth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 146–153. ACM, 2005.

Sandor Szedmak, John Shawe-Taylor, and Emilio Parado-Hernandez. Learning via linear operators: Maximum margin regression; multiclass and multiview learning at one-class complexity. Technical report, University of Southampton, 2005.