LEAR and XRCE's participation to Visual Concept Detection Task - ImageCLEF 2010

Thomas Mensink[1,2], Gabriela Csurka[1], Florent Perronnin[1],
Jorge Sánchez[1], and Jakob Verbeek[2]

[1] Xerox Research Centre Europe, Meylan, France,
`firstname.lastname@xrce.xerox.com`
[2] LEAR, INRIA Rhône-Alpes, Montbonnot, France,
`firstname.lastname@inrialpes.fr`

Abstract. In this paper we present the common effort of Lear and XRCE for the ImageCLEF Visual Concept Detection and Annotation Task. We first sought to combine our individual state-of-the-art approaches: the Fisher vector image representation, with the TagProp method for image auto-annotation. Our second motivation was to investigate the annotation performance by using extra information in the form of provided Flickr-tags.

The results show that using the Flickr-tags in combination with visual features improves the results of any method using only visual features. Our winning system, an early-fusion linear-SVM classifier, trained on visual and Flickr-tags features, obtains 45.5% in mean Average Precision (mAP), almost a 5% absolute improvement compared to the best visual-only system. Our best visual-only system obtains 39.0% mAP, and is close to the best visual-only system. It is a late-fusion linear-SVM classifier, trained on two types of visual features (SIFT and colour). The performance of TagProp is close to our SVM classifiers.

The methods presented in this paper, are all scalable to large datasets and/or many concepts. This is due to the fast FK framework for image representation, and due to the classifiers. The linear SVM classifier has proven to scale well for large datasets. The k-NN approach of TagProp, is interesting in this respect since it requires only 2 parameters per concept.

Keywords: Image Classification, Auto Annotation, Multi-Modal, Linear SVM, Fisher Vectors, TagProp

1 Introduction

In our participation to the ImageCLEF Visual Concept Detection and Annotation Task (VCDT) we focused on two main aspects. First, we wanted to investigate the effect of using the available modalities, visual (image) and textual (Flickr-tags), both at train and test time. Our second goal was to compare some of our recent techniques that potentially scale to large data sets with many concepts on the proposed task.

The VCDT is a multi-label classification challenge on the MIR Flickr dataset [5]. It aims at automatic annotation of $10,000$ test images with multiple concepts, learned from $8,000$ train images. The 93 concepts include abstract categories (like Partylife), the time of day (like day or night), persons (like no person visible, small or big group)

and quality (like blurred or underexposed). For a complete overview of the challenge see [12].

This year's challenge allowed the use of *'multi-modal approaches that consider visual information and/or Flickr user tags and/or EXIF information'*. For all images in the train and test set the original tag data of the Flickr users (further denoted as Flickr-tags) was provided. The set of Flickr-tags contains over $53,000$ different tags, from which we use a subset of most occurring tags. Also, for most of the photos the EXIF data was provided, however in our experiments we did not use this information.

In Fig. 1 an image from the database is shown, together with the Flickr-tags and the annotation concepts. We see that the tags and annotation concepts are quite complementary. While the Flickr-tags of an image corresponds to concepts which are not necessary visually perceptible (*e.g.* Australia), the image annotation system is interested in the visual concepts (*e.g.* sky and clouds).

Although the objective of a user tagging his images is different from a (visual) keyword based retrieval system, the Flickr-tags might offer useful information in the annotation task. To analyse our first aspect we have used the Flickr-tags as textual representation of an image, and conducted experiments with systems using either both modalities, or using only the visual modality. The results (see Section 4) show that indeed the Flickr-tags are complementary to the visual information. All our systems using both modalities outperform any of the visual only systems.

Concerning the second aspect, in spite of the fact that the task was relatively small especially in the number of images, we tested methods that potentially scale to large annotated data sets, *e.g.* up to hundreds of thousands of labelled images, and/or many concepts. Hence, we used image representations and classifiers which are efficient both in learning and in classifying. Efficiency includes (1) the cost of computing the representations, (2) the cost of learning classifiers on these representations, and (3) the cost of classifying a new image.

As our image representation we use the Improved Fisher vectors [13, 14], which are based on the Fisher Kernel (FK) framework [6]. The Fisher vector extends the popular bag-of-visual-words (BOV) histograms [2], by not only including word counts, but also
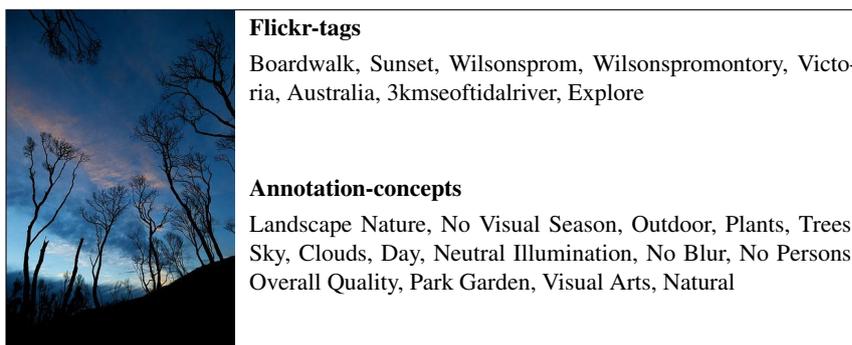

Flickr-tags

Boardwalk, Sunset, Wilsonsprom, Wilsonspromontory, Victoria, Australia, 3kmseoftidalriver, Explore

Annotation-concepts

Landscape Nature, No Visual Season, Outdoor, Plants, Trees, Sky, Clouds, Day, Neutral Illumination, No Blur, No Persons, Overall Quality, Park Garden, Visual Arts, Natural

Fig. 1. Example image with Flickr user tags, and with the ground truth annotation concepts.

additional information about the distribution of the descriptors. Due to the use of this additional information the visual code book in a FK approach could be much smaller than in the BOV approach. We use a code book of only 256 words, while a size of several thousands is common in BOV approaches. Since the size of the visual code book determines largely the computational cost for the descriptor, this makes the FK a very fast descriptor.

On the classifier part, we compare a per-keyword-trained linear Support-Vector-Machine (SVM) [16] to TagProp, a k-NN classifier with learned neighbourhood weights [4]. The training cost of a linear SVM is linear in the number of images [7, 15], therefore they can be efficiently learned with large quantities of images [10]. The advantage of the k-NN classifier is that it requires only 2 parameters per keyword, additional training for a new keyword is therefore very fast. For both classifiers we have used the same image and text representations, therefore we can fairly compare the results of the two methods.

Note that these representations and methods have shown state-of-the-art performances [4, 13, 14] on different tasks on several publicly available databases. However they were not necessarily compared or combined. The ImageCLEF VCDT challenge gave us a good opportunity to do this.

The rest of the paper is organized as follows. In Section 2 we describe the FK framework and the recent improvements on Fisher vectors. In Section 3 we give an overview of our TagProp method. Then in Section 4 we present in more detail the experiments we did, the submitted runs and the obtained results. Finally, we conclude the paper in Section 5.

2 Visual Features - the Improved Fisher vector

As image representation, we use the Improved Fisher vector [13, 14]. The Fisher vector is an extension of the bag-of-visual-words (BOV) representation, instead of characterizing an image with the number of occurrences of each visual word, it characterizes the image with a gradient vector derived from a generative probabilistic model. The gradient of the log-likelihood describes the contribution of the parameters to the generation process.

We assume that the local descriptors $X = \{x_t, t = 1 \dots T\}$ of an image are generated by a Gaussian mixture model (GMM) u_λ with parameters λ. X can be described by the gradient vector [6]:

$$G_\lambda^X = \frac{1}{T}\nabla_\lambda \log u_\lambda(X).$$ (1)

A natural kernel on these gradients is using the Fisher information matrix [6]:

$$K(X,Y) = G_\lambda^{X'} F_\lambda^{-1} G_\lambda^Y, \qquad F_\lambda = E_{x \sim u_\lambda}\left[\nabla_\lambda \log u_\lambda(x)\nabla_\lambda \log u_\lambda(x)'\right].$$ (2)

As F_λ is symmetric and positive definite, F_λ^{-1} has a Cholesky decomposition $F_\lambda^{-1} = L_\lambda' L_\lambda$. Therefore $K(X,Y)$ can be rewritten as a dot-product between normalized vectors $\mathcal{G}_\lambda$ with: $\mathcal{G}_\lambda^X = L_\lambda G_\lambda^X$. We will refer to $\mathcal{G}_\lambda^X$ as the *Fisher vector* of X.

As generative model we use a GMM: $u_\lambda(x) = \sum_{i=1}^{M} w_i u_i(x)$, with parameters $\lambda = \{w_i, \mu_i, \Sigma_i, i = 1 \dots M\}$. Gaussian u_i has mixture weight w_i, mean vector μ_i, and covariance matrix Σ_i. We assume diagonal covariance matrix Σ_i and denote the variance vector by σ_i^2. Let $\mathcal{G}_{\mu,i}^X$ (resp. $\mathcal{G}_{\sigma,i}$) be the normalized gradient vectors with respect to the μ_i (resp. σ_i) of Gaussian i. The final gradient vector $\mathcal{G}_\lambda^X$ is the concatenation of the $\mathcal{G}_{\mu,i}^X$ and $\mathcal{G}_{\sigma,i}^X$ vectors for $i = 1 \dots M$, and is therefore $2MD$-dimensional.

The Improved Fisher vector [14], takes the Fisher vector as described above and adds L2 normalization and Power normalization, both described in details below.

2.1   L2 normalization

It has been shown that the Fisher vector approximately discards image-independent (*i.e.* background) information [14]. However the vector depends on the proportion of image-specific information w.r.t. to the proportion of background information. We use the L2 norm to cancel this effect.

According to the law of large numbers Eq. 1 can be approximated as: $G_\lambda^X \approx \nabla_\lambda \int_x p(x) \log u_\lambda(x) dx$. Assume that p is a mixture containing a background component (u_λ) and an image-specific component (with image-specific distribution q), and let ω denote the mixing weight:

$$G_\lambda^X \approx \omega \nabla_\lambda \int_x q(x) \log u_\lambda(x) dx + (1 - \omega) \nabla_\lambda \int_x u_\lambda(x) \log u_\lambda(x) dx. \qquad (3)$$

Since the parameters λ are estimated with a Maximum Likelihood approach (*i.e.* to maximize $E_{x \sim u_\lambda} \log u_\lambda(x)$), the derivative of the background component approximates zero. Consequently, the FV equals $G_\lambda^X \approx \omega \nabla_\lambda \int_x q(x) \log u_\lambda(x) dx$, it focuses on the image-specific content, but depends on the proportion of image specific component ω.

Therefore, two images containing the same object but at different scales will have different signatures. To remove the dependence on ω, we L2-normalize the vector G_λ^X or equivalently $\mathcal{G}_\lambda^X$.

2.2   Power normalization

The Power normalization is motivated by an empirical observation: Fisher vectors become sparser as the number of Gaussians increases. Because fewer descriptors x_t are assigned (with a significant probability) to each Gaussian, and the derivative of a Gaussian without assigned descriptors is zero. Hence, the distribution of features in a given dimension becomes more peaky around zero, as shown in Fig 2.

Linear classification requires a dot-product kernel, however the L2 distance is a poor measure of similarity on sparse vectors. Therefore we "unsparsify" the vector z by using:

$$f(z) = \text{sign}(z)|z|^\alpha, \qquad (4)$$

where $0 \le \alpha \le 1$ is a parameter of the normalization. The optimal value of α may vary with the number M of Gaussians in the GMM. Earlier experiments have shown that $\alpha = 0.5$ is a reasonable value for $16 \le M \le 512$, so this value is used throughout the experiments. In Fig 2 the effect of this normalization is shown.
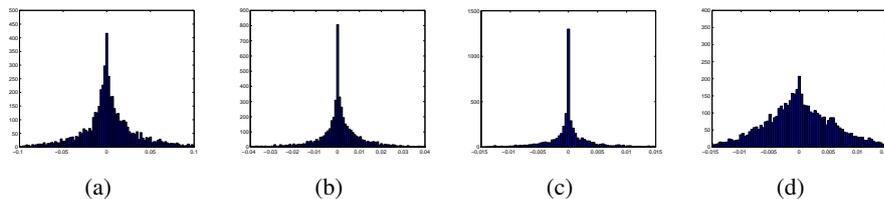
Fig. 2. Distribution of the values in the first dimension of the L2-normalized Fisher vector. (a), (b) and (c): resp. 16 Gaussians, 64 Gaussians and 256 Gaussians with no power normalization. (d): 256 Gaussians with power normalization ($\alpha = 0.5$). Note the different scales. All the histograms have been estimated on the 5,011 training images of the PASCAL VOC 2007 dataset.

When combining the power normalization and the L2 normalization, we apply the power normalization first and then the L2 normalization. We note that this does not affect the analysis of the previous section: the L2 normalization on the power-normalized vectors sill removes the influence of the mixing coefficient ω.

2.3   Spatial Pyramids

Spatial pyramid matching was introduced by Lazebnik *et al*. to take into account the rough geometry of a scene [9]. It consists in repeatedly subdividing an image and computing histograms of local features at increasingly fine resolutions by pooling descriptor-level statistics. We follow the splitting strategy adopted by the winning systems of PASCAL VOC 2008 [3], and extract 8 Fisher vectors per image: one for the whole image, three for the top, middle and bottom regions and four for each of the four quadrants.

In the case where Fisher vectors are extracted from sub-regions, the "peakiness" effect will be even more exaggerated as fewer descriptors are pooled at a region-level compared to the image-level. Hence, the power normalization is likely to be even more beneficial in this case. When combining power normalization and L2 normalization with spatial pyramids, we normalize each of the 8 Fisher vectors independently.

3   Image Annotation with TagProp

In this section we present TagProp [4, 17], our weighted nearest neighbour annotation model. We assume that some visual similarity or distance measures between images are given, abstracting away from their precise definition. We proceed by discussing how to use rank based weights with multiple distances in Section 3.2 and we extend the model by adding a per-word sigmoid function that can compensate for the different frequencies of annotation terms in the database, in Section 3.3.

3.1   A Weighted Nearest Neighbour Model

In the following we use $y_{iw} \in \{-1, +1\}$ to denote whether concept w is relevant for image i or not. The probability that concept w is relevant for image i, i.e. $p(y_{iw} = +1)$,

is obtained by taking a weighted sum of the relevance values for w of neighbouring training images j. Formally, we define:

$$p(y_{iw} = +1) = \sum_j \pi_{ij} p(y_{iw} = +1|j), \tag{5}$$

$$p(y_{iw} = +1|j) = \begin{cases} 1 - \epsilon & \text{for } y_{jw} = +1, \\ \epsilon & \text{otherwise.} \end{cases} \tag{6}$$

The π_{ij} denote the weight of training image j when predicting the annotation for image i. To ensure proper distributions, we require that $\pi_{ij} \geq 0$, and $\sum_j \pi_{ij} = 1$. The introduction of ϵ is a technicality to avoid zero prediction probabilities when none of the neighbours j have the correct relevance value. In practice we fix $\epsilon = 10^{-5}$, although the exact value has little impact on performance.

The parameters of the model control the weights π_{ij}. To estimate these parameters we maximize the log-likelihood of predicting the correct annotations for training images in a leave-one-out manner. Taking care to exclude each training image as a neighbour of itself, i.e. by setting $\pi_{ii} = 0$, our objective is to maximize the log-likelihood:

$$\mathcal{L} = \sum_{i,w} \ln p(y_{iw}). \tag{7}$$

3.2   Rank-based weighting

In our experiments we use rank-based TagProp, which has shown good performance on the MIR Flickr database [17]. When using rank-based weights we set $\pi_{ij} = \gamma_k$ if j is the k-th nearest neighbour of i. This directly generalizes a simple K nearest neighbour approach, where the K nearest neighbours receive an equal weight of $1/K$. The data log-likelihood (7) is concave in the parameters γ_k, and can be maximised using an EM-algorithm or a projected-gradient algorithm. In our implementation we use the latter because of its speed. To limit the computational cost of the learning algorithm we only allow non-zero weights for the first K neighbours, typically K is in the order of 100 to 1000. The number of parameters of the model then equals K. By pre-computing the K nearest neighbours of each training image the run-time of the learning algorithm is $O(NK)$ with N the number of training images.

In order to make use of several different distance measures between images we can extend the model by introducing a weight for each combination of rank and distance measure. For each distance measure d we define a weight π_{ij}^d that is equal to γ_{dk} if j is the k-th neighbour of i according to the d-th distance measure. The total weight for an image j is then given by the sum of weights $\pi_{ij} = \sum_d \pi_{ij}^d$ obtained using different distance measures. Again we require all weights to be non-negative and to sum to unity: $\sum_{j,d} \pi_{ij}^d = 1$. In this manner we effectively learn rank-based weights per distance measure, and at the same time learn how much to rely on the rank-based weights provided by each distance measure.

In the experiments we use a fixed $K = 1000$ independently from the number of distance measures used. So the effective number of k-NN per distance measures varies. *E.g.* when two distance measures are used, we take the 500 NN per distance measure. An image might occur twice, as neighbour according to both distance measures.

3.3 Word-specific Logistic Discriminants

The weighted nearest neighbour model introduced above tends to have relatively low recall scores for rare annotation terms. This effect is easy to understand as in order to receive a high probability for the presence of a term, it needs to be present among most neighbours with a significant weight. This, however, is unlikely to be the case for rare annotation terms.

To overcome this, we introduce word-specific logistic discriminant model that can boost the probability for rare terms and possibly decrease it for frequent ones. The logistic model uses weighted neighbour predictions by defining:

$$p(y_{iw} = +1) = \sigma(\alpha_w x_{iw} + \beta_w), \tag{8}$$

$$x_{iw} = \sum_j \pi_{ij} \, p(y_{iw} = +1|j), \tag{9}$$

where $\sigma(z) = (1+\exp(-z))^{-1}$ is the sigmoid function, and x_{iw} is the weighted nearest neighbour prediction for term w and image i c.f. Eq. 5. The word-specific models adds two parameters per annotation term.

In practice we estimate the parameters $\{\alpha_w, \beta_w\}$ and π_{ij} in an alternating fashion. For fixed π_{ij} the model is a logistic discriminant model, and the log-likelihood is concave in $\{\alpha_w, \beta_w\}$, and can be trained per term. In the other step we optimize the parameters that control the weights π_{ij} using gradient descent. We observe rapid convergence, typically after alternating the optimization three times.

4 ImageCLEF Experiments

In this section we describe the experiments for the VCDT. We evaluate the performance of systems using the textual and visual modality and compare them to visual-only systems. Also, we investigate the performance of per-keyword-trained SVMs compared to TagProp. See Table 1 for an overview of our submitted runs.

4.1 Dataset and Features

The dataset of this year's ImageCLEF VCDT was the MIRFlickr dataset [5, 12]. In contrast to last year, there were more concept classes (93) and the training set was extended to $8,000$ images. Also, in the 'multi-modal' approach it was allowed to use the provided textual 'Flickr-tag' information during both the training phase and test phase.

Features We extract our low level visual features from 32×32 pixel patches on regular grids (every 16 pixels) at five scales. Besides using 128-D SIFT-like Orientation Histograms (ORH) descriptors [11], we also use simple 96-D colour features (COL) in the experiments. To obtain the latter, a patch is subdivided into 4×4 sub-regions (as for the SIFT descriptor) and in each sub-region the mean and standard deviation for the three R, G and B channels are computed. Both SIFT and colour features are reduced to 64 dimensions using Principal Component Analysis (PCA).

Table 1. Overview of the submitted runs

Name	Modality	Nr Features	Remark
SVM	Mixed	7	Equally weighed late fusion
SVM	Mixed	7	Equally weighed early fusion
SVM	Visual	6	Equally weighed late fusion
SVM	Visual	6	Equally weighed early fusion
SLR	Visual	6	Equally weighed late fusion
TagProp	Mixed	7	
TagProp	Mixed	3	Visual distance summed over three spatial layouts
TagProp	Visual	6	
TagProp	Visual	2	Visual distance summed over three spatial layouts

In all our experiments, we use GMMs with $M = 256$ Gaussians to compute the Fisher vectors (referred also to as FV in which follows). The GMMs are trained using the Maximum Likelihood (ML) criterion and a standard Expectation-Maximization (EM) algorithm.

We extracted visual features using three spatial layouts (1×1, 2×2, and 1×3) as described in Section 2.3. The dimensionality of each FV is $M \times (2 * 64)$, since we take the derivative w.r.t. to mean and (diagonal) covariance. For each layout the component Fisher vectors were simply concatenated (*e.g.* 3 FVs in the 1×3 layout).

In some of the experiments we also use textual information (here the Flickr-tags). As textual representation for an image we use the binary absence/presence vector of the 698 most common tags among the over 53.000 provided Flickr-tags. We required each tag to be present in both the train-set and test-set, and for each tag to occur at least 25 times. This binary feature vector for each image i, is L2 normalized (denoted by $\mathbf{t}_i$). The tag-similarity s_{ij}^T between the tags of image i and image j is the dot-product: $s_{ij}^T = \mathbf{t}_i \cdot \mathbf{t}_j$.

4.2   SVM Experiments

In these experiments we wanted to investigate on one hand the effect of using both visual and textual modalities, and on the other hand the different fusion techniques (*early* and *late*) in this context. Since we use the FV representation, with the corresponding dot-product similarities, we use linear SVM's for all experiments. In all our experiments, we used the LIBSVM package [1] with $C = 1$ (some preliminary cross-validation results have shown this is a reasonable choice for this task).

Late Fusion   For the late fusion experiments we have learned for each concept a classifier per low level feature (FV-ORH, FV-COL) and per spatial-layout (1x1, 2x2, 1x3) leading to 6 visual classifiers per concept. In additional, we trained a classifier per concept on the textual features ($\mathbf{t}_i$). The scores of the Late Fusion SVM are obtained by

averaging the scores of the individual classifiers with equal weights. For the mixed modality we average over 7 scores, and for the visual-only over 6 scores.

We have also included a visual-only late fusion experiment using Linear Sparse Logistic Regression (SLR) [8], instead of SVM. SLR is a logistic regression classifier with a Laplacian prior. It uses the log-loss (instead of the hinge loss), and the probabilistic output might be more interpretable. Nevertheless, on all the measurements the corresponding SVM outperformed the SLR run (see Table 2).

Early Fusion For the early fusion experiments we have to concatenate the feature vectors. Since we use the dot-product kernel $K_d(i,j)$, concatenation of feature vectors is equivalent to the Early Fusion kernel: $K_{\mathrm{EF}}(i,j) = \sum_d K_d(i,j)$. We learn one SVM per concept using this kernel. We have experimented with visual-only ($d = \{1, \ldots, 6\}$) and mixed modality ($d = \{1, \ldots, 7\}$) classifiers.

Scoring Note that only the final scores (after either late or early fusion) were normalized to be between 0 and 1, as required. We defined our confidence score as: $\bar{x} = (x - \min(X))/(\max(X) - \min(X))$. This normalization does not affect the ordering, and therefore does not influence the per concept evaluation. The threshold for the binary decision (for per image evaluation) was set to 0 on the original scoring function x.

4.3 TagProp Experiments

Concerning TagProp we wanted to investigate on one hand the performance improvement by using the textual modality, and on the other hand the performance difference between SVM and TagProp using the FV representation. Therefore we have used exactly the same features, and distance measures, in these experiments as in the previous section. We have followed the word-specific rank-based TagProp, as described in Section 3. For all experiments we have used $K = 1000$, which is a good choice on this dataset as shown in [17].

We have ran two different sets of experiments using TagProp, one with and one without combining the spatial-layouts. When combining the different spatial-layouts we sum over the kernels $K_d(i,j)$ of the three spatial layouts to compute a single FV-ORH and a single FV-COL kernel. This is equivalent to early fusion of the spatial layout vectors. Using these combined visual kernels reduces the number of similarities used in TagProp, therefore effectively more neighbours per similarity are used, which might result in a better set of nearest neighbours. The 3rd (resp 7th) feature (see Table 1) is the textual kernel based on $\mathbf{t}_i$. To obtain $K = 1000$ nearest neighbours from D different similarity measures, we select from each similarity measure the $K_d = \mathrm{ceil}(K/D)$ neighbours, and concatenate those into $K = \{K_1, \ldots, K_D\}$.

The output of TagProp is a probability value, therefore we use it directly as the confidence score. For the binary decision scores we use a threshold of .5.

4.4 Analysis of the Results

Performance evaluation To determine the quality of the annotations five measures were used, three for the evaluation per concept and two for the evaluation per photo. For the

Table 2. Overview of the performance of the different submissions. For reference we have included the best scoring (according to mAP) results of several competitors.

Run	Modality	mAP	EER	AUC	F-ex	OS
SVM Early Fusion	V&T	**45.5**	**23.9**	**82.9**	65.5	**65.6**
SVM Late Fusion	V&T	43.7	24.3	82.6	62.4	63.7
TagProp CVD D3	V&T	43.7	24.5	82.4	60.1	41.1
TagProp D7	V&T	43.5	24.6	82.1	60.2	41.1
UvA MKL Mixed Mixed	V	*40.7*	*24.4*	*82.6*	**68.0**	59.1
SVM Late Fusion	V	39.0	25.8	80.9	62.7	63.8
SVM Early Fusion	V	38.9	26.3	80.5	63.9	*64.5*
SLR Late Fusion	V	37.1	26.1	80.6	60.0	58.2
TagProp CVD D2	V	36.4	27.3	79.3	58.0	38.5
TagProp D6	V	36.2	27.5	78.7	58.2	38.7
HHI S-IQ	V	34.9	28.6	78.2	62.8	63.6
IJS run1	V	33.4	28.1	78.8	59.6	59.5
MEIJI text and visual words	V&T	32.6	35.9	63.7	57.2	36.6
CNRS Mean Score 50	V&T	29.6	35.2	70.2	35.1	39.1

evaluation per concept the mean Average Precision (mAP), the equal-error-rate (EER), and the area-under the curve (AUC) are used, using the confidence scores. For the evaluation per photo the example-based F-Measure (F-ex) and the Ontology Score with Flickr Context Similarity cost map (OS) are used, which uses the binary annotation scores. More details on these measures can be found in [12].

Overview of Results In Table 2 we list the performance of our submitted runs and the highest scoring competitors, sorted on the mAP value. In Fig. 3 we show individual concept-based comparison of different algorithms, see also the caption for more details. From these results we can deduce that:

– All our approaches using visual and tag features outperform any of the visual-only approach. The performance is increased in the order of $5 - 8\%$ in mAP.
– While early fusion outperforms late fusion when we use the textual feature, there is no clear winner for the visual-only classifiers. The reason might be that the textual information is more complementary, while there is more redundancy between the different visual features.
– Combining the spatial-layout features into a single similarity (used in TagProp) gives slightly better results. This might be due to the fact that effectively more neighbours per similarity measure are used.
– While linear-SVM classifiers outperform TagProp, the performance is quite similar, especially for the mixed modality approach. The latter might be due to the weights TagProp learns for the two modalities, while the SVM uses an equal weighting. This conclusion confirms the observations made in [17] using a different set of features.
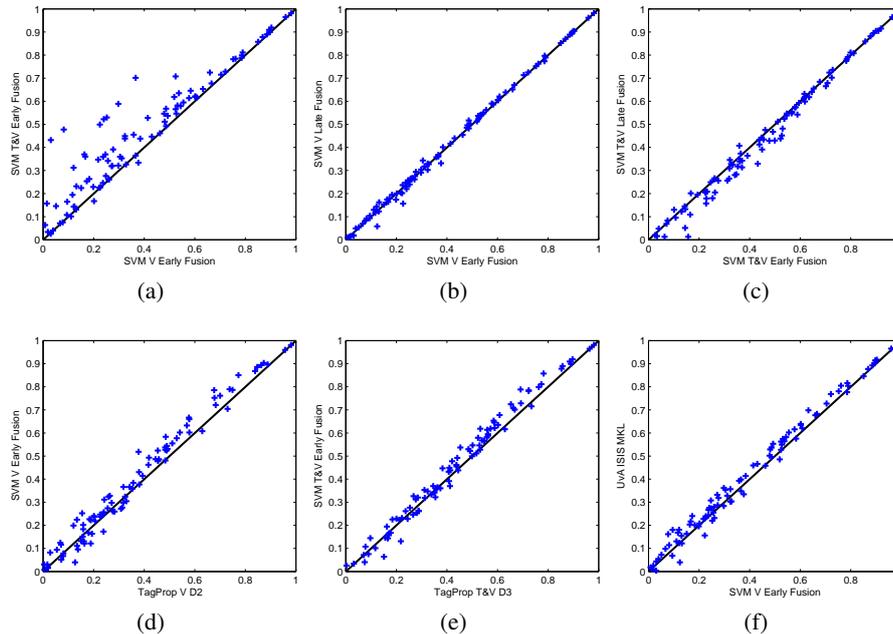
Fig. 3. Comparisons of different submissions, in each figure the AP of each concept is plotted. Plot (a) shows the performance of the best scoring SVM classifier (V&T) versus the visual only SVM. Plot (b) and (c) compares the early version late fusion SVM's. Plot (d) and (e) compares TagProp versus the early fusion SVM's. Plot (f) shows the performance of the best visual submission (UvA-MKL) versus our best visual only SVM.

 – Finally, the performance of our best visual-only classifier is close to the best scoring visual-only UvA-MKL classifier, and we are using a fast image representation with linear-SVMs.

5   Conclusions

Our goal for the ImageCLEF VCDT 2010 challenge was to take advantage of the available textual information. The experiments have shown that **all** our methods combining visual and textual modalities outperform the best visual only classifiers. Our best scoring classifier obtains 45.5 % in mAP, about 5% higher than the best visual-only system.

Besides we have compared two different approaches, linear SVM classifiers versus TagProp (a k-NN classifier). The results show that the SVM approach is superior to TagProp, but TagProp is able to compete. We believe that both these methods allow for learning from datasets with large number of images and/or concepts. Linear-SVMs have proven to scale to very large quantities images. TagProp is especially interesting for cases with many concepts and partially labelled datasets.

References

1. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
2. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV SLCV Workshop (2004)
3. Everingham, M., Gool, L.V., Williams, C., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results (2008)
4. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In: ICCV (2009)
5. Huiskes, M., Lew, M.: The MIR Flickr retrieval evaluation. In: ACM MIR (2008)
6. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: NIPS (1999)
7. Joachims, T.: Training linear svms in linear time. In: KDD (2006)
8. Krishnapuram, B., Carin, L., Figueiredo, M., Hartemink, A.: Sparse multinomial logistic regression: Fast algorithms and generalization bounds. PAMI 27(6), 957–968 (2005)
9. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)
10. Li, Y., Crandall, D., Huttenlocher, D.: Landmark classification in large-scale image collections. In: ICCV (2009)
11. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2) (2004)
12. Nowak, S., Huiskes, M.: New strategies for image annotation: Overview of the photo annotation task at ImageCLEF 2010. In: Working Notes of CLEF (2010)
13. Perronnin, F., Dance, C.: Fisher kernels on visual vocabularies for image categorization. In: CVPR (2007)
14. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: ECCV (2010)
15. Shalev-Shwartz, S., Singer, Y., Srebro, N.: Pegasos: Primal estimate sub-gradient solver for SVM. In: ICML (2007)
16. Vapnik, V.: The Nature of Statistical Learning Theory. Springer Verlag (1995)
17. Verbeek, J., Guillaumin, M., Mensink, T., Schmid, C.: Image annotation with tagprop on the mirflickr set. In: ACM Multimedia Information Retrieval (mar 2010)