Probabilistic Approaches to Video Retrieval
The Lowlands Team at TREC VID 2004

Tzvetanka Ianeva[◇,*], Lioudmila Boldareva[†], Thijs Westerveld[‡], Roberto Cornacchia[‡], Djoerd Hiemstra[†], and Arjen P. de Vries[‡]

[◇]Departament d'Informàtica
Universitat de València
València, Spain
tzveta.ianeva@uv.es

[†]Centre for Telematics and Information Technology
University of Twente
Enschede, The Netherlands
{l.boldareva,d.hiemstra}@utwente.nl

[‡]Centrum voor Wiskunde en Informatica (CWI)
Amsterdam, The Netherlands
{thijs,roberto,arjen}@cwi.nl

Abstract

Our experiments for TRECVID 2004 further investigate the applicability of the so-called "Generative Probabilistic Models to video retrieval". TRECVID 2003 results demonstrated that mixture models computed from video shot sequences improve the precision of "query by examples" results when compared to models computed from keyframes. This year, we extended these video models to capture more complex temporal events, by building generative probabilistic models from the shots using the full covariance matrix instead of a diagonal covariance matrix. Also, we improved upon the models of the associated textual data (from ASR and OCR) by introducing a multi-layered hierarchical language model. Finally, we tried to take advantage of the information in the audio channel. In the interactive experiments, we experimented with the automatic selection of the media representation (visual, textual or combined) that is most informative for answering the user's information need.

Description of the runs

Visual information All runs with visual information only.

`LL-M-dyn-sel-RR` dynamic probabilistic model, examples selected by user, round-robin (RR) combination strategy;

`LL-M-stat-sel-RR` same as above but using the static probabilistic model to verify the superiority of the dynamic approach;

`LL-M-dyn-sel-CMS` dynamic probabilistic model, examples selected by user, CMS (average sum of the scores) combination strategy to see whether round-robin combination can be improved upon.

Conclusion: dynamic is better than static, and round-robin combination approach is better than CMS.

Ordering of the examples For visual information only: what is the influence of the ordering of the examples?

`LL-F-dyn-allvidim-RR` dynamic probabilistic model, all examples reordered to move video shots to the front;

[*] supported by Valencian government grants GV CTBPRR/2002/21 and CTESPRR/2004/003

LL-F-dyn-allimvid-RR same as above, but with the original ordering (i.e. images first);

LL-F-stat-allvidim-RR static probabilistic model, all examples reordered to move video shots to the front;

LL-F-stat-allimvid-RR same as above, but with the original ordering (i.e. images first).

Conclusion: for dynamic, video examples give best results; for static, image examples work best.

Textual information ASR = automatic speech recognition, OCR = optical character recognition, full = all word tokens automatically selected from text topics descriptions, man word tokens manually selected from text topics descriptions, RR = round-robin combination

LL-F-ASR-full language model using automatic ASR and text from topic descriptions;

LL-M-OCR-full same as above using OCR instead of ASR transcripts;

LL-M-ASR-OCR-full same as above with merged ASR and OCR text data;

LL-M-ASR-man language model using ASR transcripts and words from the topics selected manually.

Conclusion: ASR+OCR improves upon ASR; manual ASR improves upon automatic ASR.

Combining visual and textual

LL-M-dyn-sel-ASR-RR dynamic probabilistic model and ASR, the user-selected examples RR combined;

LL-M-stat-sel-ASR-RR same as above but with static model;

LL-M-dyn-sel-CMS-ASR dynamic probabilistic model, user-selected examples, CMS combination then ASR;

LL-F-dyn-all-vidim-ASR-RR dynamic probabilistic model and ASR, all examples RR with videos before images;

LL-F-stat-all-vidim-ASR-RR same as above but with static model;

LL-F-dyn-all-imvid-ASR-RR dynamic probabilistic model and ASR, all examples RR with images before videos;

LL-F-stat-all-imvid-ASR-RR same as above but with static model.

Conclusion: Combining visual and textual information gives better results than either on its own.

Combining visual, textual and audio

LL-M-dyn-sel-ASR-audio-RR dynamic probabilistic model, audio, and ASR, user-selected examples RR combined;

LL-M-stat-sel-ASR-audio-RR same as above but with static model.

Conclusion: Our audio models do not contribute to improving upon visual-textual runs.

Interactive retrieval

LL-I-base-V using pre-computed Gaussian mixture models with ALA as a distance between key frames;

LL-I-comb-TV combination at search-time: visual-based (same as in LL-I-base-V) and ASR-based score (similar to LL-M-ASR-man) with equal weights;

LL-I-comb-T-V-TV either visual-based, ASR-based or their combination as above.

Conclusion: Including text-based scores improves upon using visual information only.

1 Introduction

Our video track results last year [11] demonstrated that even though the ASR run is *usually* better than the visual run, matching against both modalities ensures robustness against choosing the wrong content representation. For the same reason, using multiple visual examples to represent the information need is preferable over using a single designated example only. Following our positive findings last year, we experiment this year with merging knowledge from speech, vision, and audio. Also to take a bigger advantage of information from all available sources, we first improved and optimized the models used for each modality individually.

For this year's TRECVID workshop, we performed experiments to investigate the following research questions:

- how to represent the different modalities in probabilistic models?
- in an automatic setting, how to combine results obtained from different modalities?
- in an interactive setting, how to choose the 'correct' combination of verbal and visual modalities at search-time?
- how to combine results obtained from various query examples?

The paper is organized as follows. First, we describe the improvements made upon the individual media representations. For a better model of the visual data, we extended the dynamic models we used last year (described in detail in [7]) to capture also non-linear spatio-temporal information. The new dynamic retrieval models, aimed to improve visual performance, are described in 2.1.

In order to make a better use of the available textual information, we experimented with ASR, OCR and their combination, from which we create the hierarchical language models described in Section 2.3. Finally, we constructed generative audio models in the same way we build image models. Section 2.2 explains the details.

In the interactive retrieval setting (Section 2.4) we used text-based similarity between shots to combine with the visual-based similarity, as this is known to have positive effect on retrieval. We also made an attempt to track the progress during the retrieval session by monitoring marginal entropy and, based on that information, to alternate appropriately between similarity measures based on verbal information, visual information, or a combination of the two.

The presentation of the separate models is followed in Subsection 4.2 by a discussion of the experiments we did for combining different examples. Section 6 presents the combination of different modalities, i.e., experiments we did to combine textual, visual and audio information. The results for our interactive experiments are discussed in Section 7.

2 Retrieval Model

In the visual *static* model, keyframes images (w_i) are modeled as mixtures of Gaussians with a fixed number of components C:

$$P(\boldsymbol{x}|\omega_i) = \sum_{c=1}^{N_C} P(C_{i,c})\ \mathcal{G}(\boldsymbol{x}, \boldsymbol{\mu}_{i,c}, \boldsymbol{\Sigma}_{i,c}), \qquad (1)$$

where N_C is the number of components in the mixture model, $C_{i,c}$ is component c of class model ω_i and $\mathcal{G}(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the Gaussian density with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$:

$$\mathcal{G}(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}|}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}, \qquad (2)$$

where n is the dimensionality of the feature space and $(\boldsymbol{x}-\boldsymbol{\mu})^T$ is the matrix transpose of $(\boldsymbol{x}-\boldsymbol{\mu})$. For more details see [8, 10, 9].

As this model represents keyframes instead of complete shots, we experimented at TRECVID2003 with a new, *dynamic* retrieval model. The dynamic model is again a Gaussian Mixture Model, which extends the static model with the temporal dimension. Dynamic models are computed from one-second sequences around the keyframe, trained on samples (8 by 8 pixel blocks) described by their DCT coefficients and spatio-temporal position. Assuming a diagonal covariance matrix, resulting models are trained using standard EM [3].

The main advantages of dynamic models are:

- the reduced dependency on choosing an appropriate keyframe
- capability to capture spatio-temporal information as appearance and disappearance of objects

- integrated with information from ASR, they outperform ASR only results.

More details can be found in [7, 11].

To be able to describe more complicated spatiotemporal events we extended and optimized the dynamic model to a new dynamic model for TRECVID 2004, described in next section.

2.1 New Dynamic Model

Last year, we constrained the covariance matrices in the dynamic model to be diagonal. This constraint reduces the degrees of freedom for the covariance matrix of a single Gaussian from $15 \cdot 14/2 = 104$ (to describe a symmetric matrix of dimension 15×15) to 15 for a diagonal matrix. This makes the EM learning algorithm more robust and faster. With a diagonal covariance matrix we can differentiate between static and moving blobs and we can capture appearance and disappearance of objects. However, we cannot distinguish "jumping objects" from smooth motion since a single Gaussian with diagonal covariance does not model dependencies between time and space. Going to full covariance, we get off-diagonal covariance parameters $\Sigma_{x,t}$ and $\Sigma_{y,t}$ that capture dependencies between location and time of the object. Not only do we get this way a measure of the magnitude of the motion but also of direction. By using a mixture of several Gaussians, we can produce a piecewise linear approximation of nonlinear trajectories.

With the help of a Cholesky decomposition of the covariance matrix, the iterations of the EM algorithm are for full covariance matrices about as fast as for diagonal ones; however, since there are more degrees of freedom more iterations are required for the error to converge.

In our implementation we had to deal with numerical problems such as covariance matrices turning out not to be positive semidefinite because of rounding errors. As is done in the literature, we fix this by correcting very small false negative eigenvalues; since the magnitude of the change is small, the resulting positive semidefinite matrix is close to the unfixed matrix and hence to the exact result of our computations.

2.2 Audio

To model audio, we use the same models we use to model images or video: Gaussian mixture models. The only difference with the visual retrieval models described at the beginning of this section lies in the type of features that the models are based on. While the visual models are based on DCT coefficients describing color and texture information, the audio models are based on MFCCs, describing the acoustic energy in different frequency bands. Each shot is cut into small audio frames and for each frame, we compute the MFCCs. The position of the audio frame within the shot is not modelled, thus allowing similar sounds at the beginning and end of the shot to be modelled by a single Gaussian component.

2.3 ASR/OCR

We used two sources of automatically generated textual information: One based on the results of automatic speech recognition (ASR) provided by Limsi [5], and the other based on the results of optical character recognition (OCR) provided by CMU [6]. The approach uses a hierarchical language model. We model video as a sequence of scenes, each consisting of a sequence of shots. The generative model mixes four different levels of the hierarchy: shots, scenes, complete videos, and the total collection. Given a query with n terms $\boldsymbol{q} = (q_1, q_2, \ldots, q_n)$, the score of a shot ω_i is defined as:

$$\begin{aligned}\text{score}(\omega_i) \;=\; \textstyle\sum_{j=1}^{n} \log\Big(&\alpha P(q_j|\text{Shot}_i) + \beta P(q_j|\text{Scene}_i)\\ &+ \gamma P(q_j|\text{Video}_i) + \delta P(q_j|\text{Collection})\Big)\end{aligned}$$

where $\alpha + \beta + \gamma + \delta = 1$ and Shot_i, Scene_i and Video_i are respectively the shot, scene and video to which ω_i belongs. The main idea behind this approach is that a good shot contains the query terms, and is part of a scene having more occurrences of the query terms, which is part of a video having even more occurrences of the query terms. Also, by taking account of the text in scenes in the ranking function, we hope to retrieve the shot of interest, even if the video's speech describes it just before it begins or just after it is finished. Because we do not have scene boundaries, we assume pragmatically that each sequence of 5 consecutive shots forms a scene. The features in the text model are simply the word tokens from the transcript ASR transcript or OCR results. We estimate the probabilities $P(q_j|X)$ as the number of tokens q_j in X divided by the length of X.

Table 1: Average precision results on TRECVID 2003

	optimum at			average
training	α	β	γ	precision
2002 no video model	0.21	0.09	0.0	0.133
2003 no video model	0.40	0.40	0.0	0.134
2003 + video model	0.40	0.40	0.02	0.148

We used the TRECVID-2003 video search collection to find the optimal values for the mixing parameters: $\alpha = 0.4$, $\beta = 0.4$, $\gamma = 0.02$ (and therefore $\delta = 0.18$). Table 1 shows the average precision results on the TRECVID-2003 video collection when:
a) trained on the 2002 data, without the video model; (here we used the optimum values from the 2003 system [11]; last year we did not include the video model, so $\delta = 0$)
b) trained on the 2003 data, without the video model;
c) trained on the 2003 data, with the video model.
The differences in average precision between training on the 2002 data and training on the 2003 data are negligible. If we however include the video model in the ranking function, results on the training data go up from 0.134 to 0.148, which is significant following a paired sign test at the 95% level.

2.4 Interactive retrieval

2.4.1 Procedure

The model for interactive retrieval is similar to the one used in the previous TRECVID experiment [11]. After seeing the topic description, the user may post a text query to initiate browsing through the collection. To make interactive experiments more controllable, the text query made by the users was saved but not used. Instead, the output of the manual ASR run (see Section 2.3) determined the initial ranking of shots. Based on this ranking, the key frames of n best scoring shots are presented to the user. Each key frame is accompanied with k key words that most likely belong to the corresponding shot, according to the language model based on ASR.

Upon studying the displayed key frames and their keywords, the user could mark certain key frames as “good” or “bad” examples, thus providing relevance feedback. The user could also “save” shots that would satisfy to the search task that she/he is performing, before proceeding to the next iteration. Saved shots are automatically marked as good examples.

2.4.2 Iterative score re-computation

The feedback given by the searcher, denoted here $F(\boldsymbol{x}_1 \dots \boldsymbol{x}_n)$, is used to update the scores for unseen objects, assuming conditional independence between the marked key frames $\boldsymbol{x}_1 \dots \boldsymbol{x}_n$ given the hypothetical target ω_i:

$$P(\omega_i|F(\boldsymbol{x}_1 \dots \boldsymbol{x}_n)) \propto P(\boldsymbol{x}_1 \dots \boldsymbol{x}_n|\omega_i) \;=\; \prod_{s=1}^{n} P'(\boldsymbol{x}_s|\omega_i) \tag{3}$$

$P'(\boldsymbol{x}_j|\omega_i)$ denotes the probability that $\boldsymbol{x}_j$ is selected by the user as a positive example while ω_i can satisfy the user's information need.[1]It is computed from the distribution of all pairs $P(\boldsymbol{x}|\omega_i)$ by fitting it onto the Normal distribution. More detail about how $P'(\boldsymbol{x}_j|\omega_i)$ are computed can be found in [2].

Probabilities $P'(\boldsymbol{x}_j|\omega_i)$ are pre-computed at indexing time. Only the values beyond a certain threshold are actually stored in the index called association matrix. The values not in the index are substituted with a smoothing constant $\bar{p}$. Such set-up enables fast access to the data, but does not impair retrieval quality [2].

2.4.3 Models to compute similarity

We used the ASR-only language model described above to compute pairwise similarity values to be used for indexing the text modality.

[1] I.e., that the user associates $\boldsymbol{x}_j$ with ω_i.

To index the visual part of the videos, we used our static Gaussian mixture models, but with the asymptotic likelihood approximation [8] as a distance measure. This choice has been made to speed up the preprocessing (building the matrix with pairwise similarities).

2.4.4 Combining modalities in interactive search

For verbal and visual modalities two separate association matrices are computed. Combining them at search time is performed as follows: We assume that the user selects a key frame either because of its appearance or because of the key words that belong to the corresponding shot. Because the association matrix contains probabilities, the scores computed for each modality are on the same scale, and are simply added up with equal weights.

For one run, the score was updated using either of the two matrices, or their combination as above. To determine which update strategy to use, we looked at the effect of each variant on the next iteration. From the Information theory we know that entropy of a system is a measure of the uncertainty level in it. Marginal entropy is closely associated with the discriminating power of query terms [4] and correlates with MAP [1].

As we are using a probabilistic framework for retrieval, computing marginal entropy does not take much overhead. After each iteration we compute visual-based and verbal-based scores and their combination. The distribution that yielded the largest decrease in entropy was used to proceed with the next iteration.

3 Experimental Setup

3.1 Building Models and Queries

To index the search test collection for each shot we build a dynamic, static, language and audio model.

3.2 Singularity Problems

By maximizing the likelihood (equivalent to minimizing the error function) of the parameters for the given data set we iteratively fit the model according to the set of data. But there exist parameter values for which the likelihood goes to infinity. Theoretically this can be seen if we set $\mu = x$ in Eq. (2) and then letting $\Sigma \to 0$. This is the "singular solutions problem." In practice what is happening is that sometimes one of the components collapses onto a very small region in the feature space and the component explains nothing except for this particular region. Last year we worked around this problem by setting the prior probability of components with a covariance smaller than some threshold to zero. Effectively, this means we ignored these components during retrieval. This year we used a more sophisticated approach to improve retrieval performance.

Several techniques have been proposed to deal with singularity problems. On a subset of the TRECVID2003 data, we experimented with the following approaches, assuming a fixed number of components. When one of the variance parameters shrinks to a very small value during the EM algorithm

- the covariance matrix of the corresponding Gaussian is reset to the initial covariance matrix
- corresponding Gaussian is replaced with one having a larger width, i.e., the covariance matrix continues being small but not *too* small
- we set the covariances of *all* components to the mean of the current covariances

The last method, dubbed "equal" by us, turns out to yield the best results: the error function converges faster and the models fixed in this way show higher performance in retrieval.

Intuitively, these pathologically small covariances may be an indicator that our preset fixed number of components (eight in this year's experiments) is too high. The dynamic model is always built from a one-second sequence around the keyframe. Depending on the chosen keyframe, there may therefore be some completely black images in this sequence and components modelling such artefacts will have high likelihood because of their uniformity. Therefore we would like to discard such components from the model. Similarly, we could have the situation that the sequence to be modeled contains only few objects or regions, e.g., a jumping ball in front of a uniform background. Sometimes the aspect ratio of a movie is adapted to the aspect ratio of television, leaving black bars at the top and bottom of the picture. When analyzing what unbounded-likelihood components model, we saw that these were often thin black lines that came from such regions. A similar effect is caused by overlays in news broadcasts.

Convergence is faster and the error at the cut-off smaller if we have just the "right" number of components. Therefore we considered strategies based on reducing the number of components and the small-covariance fixes described earlier, looking for a favorable trade-off between complexity and computation speed. In the end we settled for the following strategy for TRECVID 2004: when the magnitude (measured by a combination of determinant and minimum singular value) of a covariance matrix falls below a threshold, we remove the corresponding component and recompute the priors of the other components. If this removal reduces the error, then we resume the training without that component; if, on the other hand, the removal increases the error, then we undo the removal and apply the "equal" fix method to the small covariance matrix and continue the training with all components. Of course, if more than one covariance matrix is small, we apply the same treatment to all of them.

3.3 Queries

The interactive runs only used textual queries. For automatic runs, building queries from topic descriptions is automatic. The only difference among automatic runs was whether to use the dynamic or the static model, and the order of the query examples (videos first or images first). The only manual action in constructing visual queries was selecting the set of image and video examples to be used for ranking. Textual queries were constructed automatically from the search topic for the runs, except for the run `LL-M-ASR-man`. All query examples are rescaled to at most 240×352 pixels.

4 Visual results

4.1 Performance of the dynamic model

In order to describe more complicated nonlinear temporal events this year we create generative probabilistic models from the shots using a full covariance matrix instead of a diagonal covariance matrix. In other words, the Gaussians are no longer axes-aligned. Also, we run separate queries for each example (selected by the user) and merge the results afterwards in a simple round-robin approach following the order decided by the user.

Better modeling does not necessarily imply better retrieval. Thus we also compute models from keyframes, i.e., models that only make use of static visual information and combine them same way. As expected we obtain higher map when using dynamic models.

The dynamic model represents the spatio-temporal information in the shot, as opposed to just spatial information. Thus we assume that dynamic models give matches more consistent with the visual content of the query represented by video shots rather than images. To verify this assumption, we perform two automatic runs using all examples in a round-robin fashion. The order of the examples is varied. Topics are described by examples where always image examples come before video examples. In the first run we use first video then image examples. In the second run we do not introduce any changes, i.e., we use examples as they are in the topic descriptions. In agreement with our expectations, the performance is higher when videos are used first.

Doing the same runs with the static model (which is very suitable for image retrieval) leads to the opposite result, underlining the importance of the dynamic models.

4.2 Merging Visual Run Results

Often it is impossible to find shots that are visually similar to all query examples for a given topic. The user who selects query examples to be used introduces filtering of the visually bad examples according his criteria, i.e., only excluding examples that he thinks do not represent visually well enough the information need. Thus our combining strategy: ranking the different query examples separately and combining the results afterwards in a Round Robin fashion following user's order is very appropriate for visual modality. To show the better performance of our strategy we combine the query examples in other well known as successful approach based on scores; (CMS)-ranking documents based on the mean sum of the individual scores.

5 Textual Results

To combine ASR and OCR, the available screen captions are merged with the results of speech recognition engine and further used as described in Section 2.3.

For the manual run using ASR, the words to use as a query were selected manually from the topics descriptions. In other cases the text was pre-processed automatically which included stemming and stopwords removal. The results for full and automatic runs are presented in Table 2. Although in general ASR shows higher performance than OCR, adding text from OCR turns out to be advantageous.

Run Index	**MAP**
`LL-F-ASR-full`	0.0680
`LL-M-OCR-full`	0.0046
`LL-M-ASR-OCR-full`	0.0691
`LL-M-ASR-man`	0.0760

Table 2: Mean average precision results for text-based runs

The difference, however, is insignificant at the 95% level according to both the Sign test and Wilcoxon Signed-Ranks Test. A quick glance at individual topics shows that for some topics the OCR-based model returns relevant shots not found when using only ASR. For other topics OCR returns a subset of shots that are also found by ASR-model, but very often in the OCR run they are at the top and therefore they improve the precision of the ASR run. At top 30 cut-off level, the ASR run failed to find any relevant shots in 8 topics, whereas the combined ASR-OCR model only failed to find relevant shots in 6 topics. This emphasizes usefulness of adding OCR information, because for instance an interactive retrieval system relies on relevance feedback on few shots from the top, that are useful to the user.

Since adding OCR turned out to be beneficial for retrieval, the performance of OCR-based run may further be refined by pre-processing the OCR output in order to correct obvious optical recognition errors.

6 Combining Modalities

A video retrieval system should take advantage of information from all available sources and modalities. Since in our case all modalities are modelled in a probabilistic framework, combining them is straightforward, or at least it is if we assume the modalities are independent. The independence assumption is surely debatable. If for example textual and visual information were completely independent, then using textual queries to find visual information would be useless. The fact that the textual runs are still the most successful monomodal runs shows that textual information does tell something about the visual content. Nevertheless, in last year's TRECVID [11] we saw that this naive approach of independently combining modalities could improve over mono-modal runs. This year we followed the same strategy to combine textual, visual and audio runs, simply by computing the joint probability of generating the textual query from the language models, the visual examples from the visual models, and the audio in the video examples from the audio models.

We combined automatic visual runs with the automatic ASR runs and manual visual runs (selected examples) with manual textual runs (short, modified queries). The manual combinations are in addition combined with the audio results. All combinations of visual and textual runs performed significantly better (Wilcoxon signed rank test at 95% level) than the corresponding mono-modal variants. Adding audio information however degrades results. More research is needed to improve the audio models. But also rethinking the way of combining the different modalities could be useful. For example, it could be interesting to explore ways of dropping the independence assumptions.

7 Interactive Experiments

At the moment of submission, the ranked list with the search results for interactive runs was occasionally sorted alphabetically. This consequently resulted in extremely low MAP. Post-hoc the search results were re-evaluated using the correctly sorted ranked lists, and further we report those figures.

Three users performed the three experiments, using Latin square experiment design. Time spent on one topic did not exceed 15 minutes. The users could finish the search session at any moment if they felt they found enough relevant key frames. Like last year, the video clips were not available to the users who instead observed the key frames supplied with the data.

From previous experiments we learned that in some cases users would prefer to have a possibility to get rid of series of almost identical key frames that are irrelevant. For that purpose the users were instructed to use negative relevance feedback functionality. MAP for the run that used only visual modality to update relevance scores serves as our baseline. Adding text modality significantly (Wilcoxon signed rank test at 95%, Sign test not significant) improves the visual-only run for most of the topics and result in overall better performance.

Run Index	**MAP**
`LL-I-base-V`	0.1273
`LL-I-comb-TV`	0.1900
`LL-I-comb-T-V-TV`	0.1661
`LL-I-base-T`	0.1875

Table 3: Mean average precision results for interactive runs

The model that uses marginal entropy to determine how to compute the score for the next iteration, in half of the cases did worse compared to the fixed 50/50 combination of the two modalities. The difference between these two runs is not significant at the 95% level according to both the sign test and Wilcoxon signed ranks test.

By looking at individual topics performance we found that for topic 130 the corresponding MAP are 0.7009 (fixed combination) and 0.3381 (entropy-based), a substantially large difference. Without this only outlier the difference between the two runs is small: 0.1668 for the fixed combination vs 0.1583 for the entropy-based one.

We also conducted an additional post-hoc experiment that uses text modality alone for updating scores. MAP is shown in the table in the last line. As we see, the combination of text and visual score improves both single-modality versions, but for the text-based one the increase in MAP is negligible and the difference is not significant according to both tests used.

We found however that in two thirds of the cases when adding visual-based scores to the ASR decreased MAP, the entropy-based combination performed better than the fixed combination model. This indicates potential usefulness of such approach.

The good finding is that our interactive combination strategy is in general beneficial even when one modality performs not very well. This confirms the automated experiments reported in [2]

8 Conclusions

With the full-covariance dynamic model we get more out of the visual information; computing full covariance models required us to explore solutions for numerical and performance problems.

In our combination strategy the combination of runs based on multiple modalities is successful if and only if the runs being combined each do something useful. This year once more our visual runs improve ASR. This was not the case with audio. Since this is our first attempt at integrating audio in our system, we think not the combination strategy but rather the extraction of audio descriptors and the resulting audio models are not sufficiently good yet.

Selection of good visual examples for a given topic from the user and our purely based on the ranks approach for their combination is the main cause for the success of the runs not only inside the visual modality but also when combining visual and ASR runs.

A rank-based approach for combination of visual modalities and a score-rank based approach for combining ASR with visual information are the main ingredients of our best runs.

When comparing results to those of last year, more experiments are needed. For example, in the TRECVID 2003 experiments, we used JPEG compression of query images with a quality level of 20% to match size and quality of the collections videos. Also, we applied detectors of (amongst others) anchor persons to improve upon our results. To be able to analyze correctly the results and understand better our models this year we did not use such advanced pre- and post-processing steps.

References

[1] L. Boldareva, A. de Vries, and D. Hiemstra. Monitoring User-System Performance in Interactive Retrieval Tasks. In *Recherche d'Informations Assistee par Ordinateur (RIAO 2004)*, pages 474–483, Apr. 2004.

[2] L. Boldareva and D. Hiemstra. Interactive Content-Based Retrieval Using Pre-computed Object-Object Similarities. In *International Conference on Image and Video Retrieval*, volume 3115 of *LNCS*, pages 308–316. Springer, July 2004.

[3] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, series B*, 39(1):1–38, 1977.

[4] S. Dominich, J. Goth, T. Kiezer, and Z. Szlavik. An entropy-based interpretation of retrieval status value based retrieval, and its application to the computation of term and query discrimination value. *Journal of the American Society for Information Science and Technology*, 55:613–627, May 2004.

[5] J. Gauvain, L. Lamel, and G. Adda. The LIMSI broadcast news transcription system. *Speech Communication*, 37(1–2):89–108, 2002.

[6] A. Hauptman, R. Baron, M.-Y. Chen, M. Christel, P. Duygulu, C. Huang, R. Jin, W.-H. Lin, T. Ng, N. Moraveji, N. Papernick, C. Snoek, G. Tzanetakis, J. Yang, R. Yan, and H. Wactlar. Informedia at trecvid 2003 : Analyzing and searching broadcast news video. In *TRECVID 2003 Workshop*, 2003.

[7] T. Ianeva, A. P. de Vries, and T. Westerveld. A dynamic probabilistic retrieval model. In *IEEE International Conference on Multimedia and Expo (ICME)*, 2004.

[8] N. Vasconcelos. *Bayesian Models for Visual Information Retrieval.* PhD thesis, Massachusetts Institut of Technology, 2000.

[9] T. Westerveld. *Using generative probabilistic models for multimedia retrieval.* PhD thesis, University of Twente, 2004.

[10] T. Westerveld, A. P. de Vries, A. van Ballegooij, F. M. G. de Jong, and D. Hiemstra. A probabilistic multimedia retrieval model and its evaluation. *EURASIP Journal on Applied Signal Processing*, 2003(2):186–198, 2003. special issue on Unstructured Information Management from Multimedia Data Sources.

[11] T. Westerveld, T. Ianeva, L. Boldareva, A. de Vries, and D. Hiemstra. Combining information sources for video retrieval. In *TRECVID 2003 Workshop*, 2004.