D5.5

Visual mining and linking module – v2

v 1.1 / 2017-02-10

Etienne Gadeski (CEA), Eleftherios Spyromitros-Xioufis (CERTH), Hervé Le Borgne (CEA), Symeon Papadopoulos (CERTH), Adrian Popescu (CEA), Yiannis Kompatsiaris (CERTH)

The current deliverable is a technical report accompanying the second version of the USEMP visual mining and linking modules. This deliverable is an update of D5.2 “Visual mining and linking module – v1”. Similar to D5.2, it documents the underlying principles and methodologies, the exposed functionality, the respective implementation details, and the conducted evaluation experiments. In addition, it highlights the importance of each module for the project use cases and the multi-disciplinary issues arising from their deployment.
In particular, the following modules are included and discussed: a) large-scale visual concept detection, b) private/non-private image classification, c) logo recognition and d) image based location detection. Private/non-private classification is a new module, while the others are updates of modules introduced in D5.2.

Project acronym	USEMP
Full title	User Empowerment for Enhanced Online Presence Management
Grant agreement number	611596
Funding scheme	Specific Targeted Research Project (STREP)
Work program topic	Objective ICT-2013.1.7 Future Internet Research Experimentation
Project start date	2013-10-01
Project Duration	36 months

Workpackage	5
Deliverable lead org.	CEA
Deliverable type	Prototype
Authors	Etienne Gadeski (CEA), Eleftherios Spyromitros-Xioufis (CERTH), Hervé Le Borgne (CEA), Symeon Papadopoulos (CERTH), Adrian Popescu (CEA), Yiannis Kompatsiaris (CERTH)
Reviewers	Rob Heyman (iMinds) Noel Catterall (HWC)
Version	1.1
Status	Final
Dissemination level	RE: Restricted Group
Due date	2015-11-30
Delivery date	2016-02-10 (revised 2017-02-10)

Version	Changes
0.1	ToC from CEA
0.2	Contributions from CERTH about private/non-private image classification
0.3	Contributions from CEA on concept and logo detection
0.4	First complete version available for internal review
1.0	Refinements after internal reviews
1.1	Updated version addressing comments received from the third annual review

Table of Contents

1. Introduction ... 3

1.1. Scope of the deliverable ... 3

1.2. Visual mining and linking in USEMP ... 3

1.3. Research methodology and contributions ... 4

1.4. Multidisciplinary issues ... 5

2. Large-scale visual concept detection ... 8

2.1. Related work ... 8

2.2. Method description ... 9

2.2.1. Effective training of convolutional neural networks with Web images ... 9

2.2.2. Enhancement of semantic features for image mining ... 10

2.3. Evaluation and testing ... 12

2.3.1. Evaluation of CNN training with Web images ... 12

2.3.2. Evaluation of the enhancement of semantic features ... 13

2.4. Implementation and usage ... 15

2.5. Next steps ... 15

3. Private/non-private image classification ... 16

3.1. Related work ... 17

3.2. Method description ... 18

3.2.1. *YourAlert*: A Realistic Private Image Classification Benchmark Dataset ... 18

3.2.2. Personalized Private Image Classification Models ... 19

3.2.3. Visual and Semantic Features ... 19

3.3. Evaluation and testing ... 20

3.3.1. Experimental Setup ... 20

3.3.2. Generic Image Privacy Classification Models ... 20

3.3.3. Personalized Privacy Classification Models ... 22

3.3.4. Image Privacy Exploration via SemFeat ... 23

3.4. Implementation and usage ... 24

3.5. Next steps ... 25

4. Logo detection ... 26

4.1. Related work ... 26

4.2. Method description ... 26

4.3. Evaluation and testing ... 26

4.4. Implementation and usage ... 29

4.5. Next steps 29
5. Image based location detection 30
5.1. Related work 30
5.2. Method description 30
5.3. Evaluation and testing 30
5.4. Implementation and usage 31
5.5. Next steps 31
6. Conclusions and future work 32
7. References 33

1.Introduction

This deliverable provides documentation on the second version of the prototype implementations of the USEMP visual mining and linking modules. It is an update of D5.2 “Visual mining and linking module – v1” and, since the overall objectives of the project did not change, the introductory section is largely similar to that of D5.2. This section first delineates the scope of the deliverable; it proceeds with an overview of the delivered visual mining and linking modules, underlining the differences compared to the first delivered versions in case the tools have been updated. It continues with a description of the adopted research methodology and concludes with a discussion of the multi-disciplinary issues involved in the development and deployment of the presented modules.

1.1. Scope of the deliverable

This deliverable offers documentation on the delivered prototype implementations of the second version of the USEMP visual mining and linking modules. The deliverable addresses the following objectives: a) make clear the role and usage of each module in the USEMP system, b) describe the underlying research approaches and expose a number of technical implementation details, and c) present the achieved experimental results and discuss aspects related to the deployment and integration of the modules in the system.

Although much of the deliverable content is addressed to the public community of interested researchers and practitioners, part of the discussion is dedicated to USEMP-specific aspects, contextualizing the work within the project background, work structure and plan.

1.2. Visual mining and linking in USEMP

The primary goal of building a number of visual mining and linking modules in USEMP is to endow DataBait, the USEMP tool, with the capability to **conduct inferences about an OSN users’ interests, disclosure behaviour and traits based on the visual content of the images they share**. These inferences are produced on a per-image level, but are subsequently exposed to the USEMP privacy scoring framework (documented in D6.1, D6.4), where they are aggregated and combined (together with additional inferences based on text processing – see D5.1, D5.4 – and complementary online trails and cues – discussed within WP6) to build and update rich user profiles. The types of information that can be inferred by processing users’ images include a wide variety of personal information such as:

- Interests and activities (e.g. sports, arts, activism)
- Habits (e.g. smoking, drinking)
- Favourite brands and products (e.g. mobile phones, clothes)
- Home location and list of visited places
- Social interactions (i.e. people appearing in the same image)
- Social affinities (i.e. people sharing similar content)

Due to the variety of personal information to be mined, a number of visual mining and linking modules and approaches need to be employed, and have therefore been the subject of research and development within USEMP. One of the main researched approaches is

concept detection (Section 2), i.e. the identification of entities, objects and themes of interest that are depicted in images. Concept detection is a versatile information extraction tool that can be used to detect a large variety of the aforementioned types of information, including interests, activities, and habits, contributing to the recognition of depicted scenes. Between v1 and v2, the USEMP concept detection approach was improved in two main ways: (a) showing that it is possible to train convolutional neural networks using noisy Web images and (b) using a local enhancement of semantic descriptors for image classification. A second important approach is **private/non-private image classification** (Section 3), which could alert users when sharing images that are disclosing personal information. This is an emerging research topic and the resulting automatic classification module can help users decide about the level of visibility of an image that is published on an OSN. A further approach deals with the **detection of logos and products** in images (Section 4), which is useful for detecting the association between users and brands, and can be used as a value proxy for OSN users. A final important approach is **location and POI detection** (Section 5), which attempts to estimate the location of where an image was captured (i.e. the location of the depicted scene) based on visual cues as well as with the help of matching the image of interest to other images with known location (e.g. based on Exif or OSN platform-specific metadata).

Figure 1 illustrates the foreseen usage of the USEMP visual mining and linking modules listed above in a few exemplary cases.

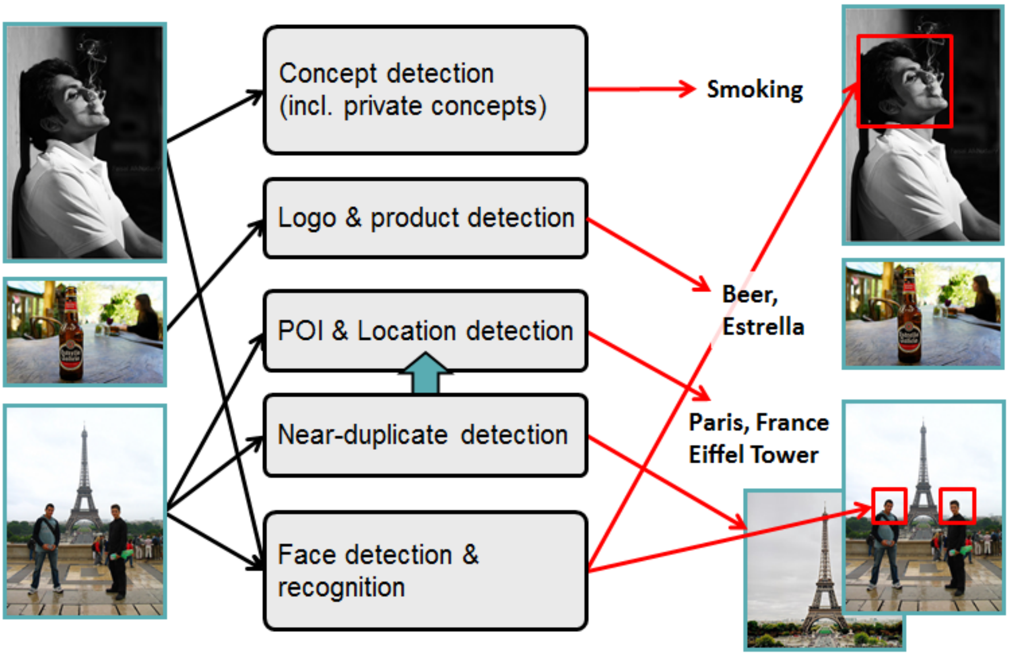

Concept detection (incl. private concepts)
Logo & product detection
POI & Location detection
Near-duplicate detection
Face detection & recognition
Smoking
Beer, Estrella
Paris, France Eiffel Tower

Figure 1. Example use and outputs of USEMP visual mining modules. To avoid overloading the image with content, only a small number of possible interconnections and outputs are presented.

1.3. Research methodology and contributions

The conducted research described in this deliverable was to a large extent shaped by the desiderata and insights coming from the USEMP disciplines (social science, legal studies, user studies, system design) as will be discussed in more detail in the next subsection. Having specified the main objectives of the visual mining and linking research in close collaboration with the above disciplines, the next step was to perform an extensive analysis of existing work on each of the studied fields (a summary of which is included in a dedicated subsection for each module). Work has followed the same methodology that was devised for D5.2. In each case, the most effective methods for the problem at hand were selected as the basis for the modules; subsequently, existing implementations of the selected methods were reused wherever possible, while in some cases development work was necessary to build the target method. Finally, to assess the reliability and quality of the prototyped solutions, they were evaluated using suitable publicly available datasets. In case no such datasets were available, new ones were created with a focus on the problems of interest. For this second version of the modules, we updated the review of state of the art techniques to keep abreast with very recent developments or proposed new when none of the existing seemed appropriate.

Although much of the work performed in this first research and development iteration relied on existing computer vision and machine learning approaches, we consider that it resulted in a number of valuable research contributions. In particular:

- In D5.2 we introduced concept-level feature representation (Semfeat) that is particularly effective both for conventional concept detection settings and, importantly, for transferring concept models to new sets of concepts. The proposed representation is grounded on state-of-the-art computer vision advances (Convolutional Neural Networks - CNN, which fall under the family of Deep Learning methods) and is tested on large-scale datasets, as well as on datasets focused on *private concepts*. The main new developments of Semfeat that are presented in this deliverable are related to: (a) effective training of convolutional neural networks with Web data and (b) the introduction of an individual image adaptation of semantic features to improve image mining results.
- A new method for private/non-private image classification is introduced here. Compared to existing methods, we perform an assessment of CNN and Semfeat features and show that they outperform more classical image descriptors, such as bags of visual words or VLAD. More importantly, we discuss limitations of generic privacy models and show the importance of user-centred feedback for the improvement of performance. A first version of a dedicated dataset was created as part for this task and it is currently enriched.
- After preliminary experiments for logo recognition presented in D5.2 that were based on a bag of visual words approach, we propose in this deliverable a Deep Learning based pipeline that clearly outperforms the previously mentioned approach both in terms of accuracy and scalability. One important conclusion here is that it is possible to learn directly from the Web, with little or no manual intervention during the creation of the training dataset.
- After testing a number of location estimation approaches in D5.2, we focused on a Deep Learning based pipeline here which is similar to the one developed for logo recognition. In order to improve scalability, focus was put on the reduction of the feature dimensionality.

1.4. Multidisciplinary issues

Although visual mining is mainly dealing with approaches from the areas of computer vision, image processing and machine learning, the presented research was considerably shaped by the rest of the USEMP disciplines, and at the same time provided actionable feedback to them. In the following, we provide a concise account of the inter-play between visual mining research and the different disciplines of the project.

D5.2 is informed by work done in WP2, WP3, WP4 and WP9 and it provides valuable input for WP6 and WP7. The legal analysis carried out in WP3, and more particularly in T3.6 which deals with coordination of legal aspects, clarified practical implications of visual content mining related to: processing of sensitive information (e.g. so that concept detection models and evaluation were tailored to effectively detecting sensitive information), copyright issues related to data used during training, ensuring that all USEMP components have clear IP rights (in case of reusing existing components). Work on trade secrets and intellectual property carried out as part of D3.2 explored the tensions between profile representations on the end-user side, within OSNs and created in USEMP and made clear the complex interplay between these actors, as well as their respective rights and obligations.

The use case analysis in D2.1 and the associated requirements defined in D2.2 served as guidelines for the implementation of technical components. In particular, the following requirements are central here:

- [SR02] "The system may be able to process the information within one second such that the user can make informed decisions on their past data without long delays. In the event data processing is to take longer, a progress bar should be presented. A maximal extent of 10 seconds will be aimed for." This requirement has strong implications in terms of processing speed for the implemented components.
- [SR04] "The system may be able to make best effort associations between data placed onto OSN(s) and the profile attributes which can be inferred from such data." This requirement is a counterpart of [SR02] that focuses on component performance, which should closely follow state of the art developments.
- [SR11] "The system may be able to get fruitful insights on how relevant a user's profile is for different stakeholders." Through inferences made by technical components, the end-users should be able to have insightful information on how her profile is seen by OSNs and, possibly, by other stakeholders.

In D4.1, a comprehensive list of social requirements was established, which offers a user-side view of functionalities that need to be implemented by USEMP tools. Of particular interest here are:

- Req. 1 asking for more transparency about privacy problems at an institutional level and notably OSNs in this context.
- Req. 2 demanding a backward link between inferences and raw data which generated them to improve the accountability and provenance of the automatic decisions made by the system.
- Req. 10 asking for a low impact on browser speed of the USEMP plug-in, a requirement which is tightly linked to [SR02] mentioned above.

The extensive market analysis done in D9.3 showed that existing privacy enhancing tools and privacy feedback and awareness tools deal mostly with volunteered and/or observed data. A strong opportunity in USEMP is to provide users with a more complete view of how their data could be handled and exploited by OSNs. Another conclusion of D9.3 is that existing content visual mining tools are not tailored for privacy enhancement and, consequently, an adaptation step is needed in order to better satisfy domain requirements.

Downstream, insights gained with D5.2 tools can be used both directly in the USEMP interface (D7.2, D7.5), and as part of the privacy scoring framework created in D6.1 and D6.4, to complement social network mining inferences. For instance, user locations can be extracted from images and can be displayed directly by the USEMP interface to inform the user about her degree of exposure on this core privacy dimension. In a more complex functioning mode, logo and product recognition from images can first be used to link the user to different brands and then can be combined with social interactions (such as links, comments, shares etc.) in order to derive a value estimate for the shared image.

2. Large-scale visual concept detection

Concept detection is the core visual mining module of USEMP because it enables the project tools to make privacy related inferences from raw images and thus build much more detailed privacy profiles. According to the insights provided by WP2, WP4 and WP6 analyses, a very large variety of concepts[1], i.e. entities, objects and themes of interest depicted in images, are illustrated in user content shared on OSNs and scalability in terms of recognizable concepts should be a core requirement, along with detection accuracy. To cope with these requirements and to keep abreast with latest developments in computer vision, the majority of concept detection experiments are performed using deep learning features. Focus is put on feature transfer from an initial training set to a larger number of concepts and on learning visual concepts from manually curated resources but also directly from the Web. This last line of research is particularly important in order to improve concept detection scalability with no or little manual effort. The results of concept detection can be either used as such or integrated with other cues (including textual and social network mining insights) to inform the users their disclosure status on OSNs. This module was already integrated in the first version of Databait and the feedback obtained during the pre-pilot is very encouraging.

2.1. Related work

As we have mentioned in D5.2, Deep Convolutional Neural Networks (CNN) (Krizhevsky et al., 2012) have emerged as an efficient end-to-end way to represent images. Image representation is no longer designed based on prior knowledge but hierarchically learned from image pixels to higher-level primitives such as edges, corner, and object parts. CNNs have recently demonstrated impressive image classification performance in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC[2]) (Russakovsky et al., 2014). Here we are particularly interested in two aspects, namely (1) the availability of datasets for efficient training of CNN architectures and (2) the exploitation of semantic features for image classification.

CNN training is usually done with manually labeled data but this approach has the obvious disadvantage that large volumes of images need to be validated by humans. In an attempt to overcome this problem, focus is put on the possibility to exploit large Web image corpora instead of manually labelled datasets. Prior work (Schroff et al., 2011), done outside the deep learning field proposes re-ranking techniques that rely on cross-validation scores of web images. A challenge related to this unsupervised re-ranking process is that it only works if noisy images are not predominant in the initial dataset. Weakly supervised methods, such as Kernel Mean Matching (Huang et al., 2006) or Transductive SVM (Sindhwani et al., 2006) exploit a reduced set of image examples to guide the re-ranking process. They constitute a good compromise between fully unsupervised learning and complete manual annotation of the training datasets and will be investigated in our context.

Image classification is predominantly performed using “bottom-up” CNN features in which concept representations are progressively abstracted from the raw content of the images that

[1] The term *concept* is an established term in the multimedia analysis and computer vision research communities and is typically associated with a topic, entity, object or theme depicted in an image.

[2] http://www.image-net.org/challenges/LSVRC/2014/ (consulted on 5/1/2015)

is given as input to the classification pipeline. In contrast, "semantic features" (Bergamo & Torresani, 2012) encode images as a series of visual concepts that are fed into the pipeline. In D5.2, we have shown that semantic feature performance is improved if sparsification is applied. However, a fixed sparsification threshold was applied to all images, regardless of their visual complexity. A related problem is that of objects appearing at different scales in the image (Li et al., 2010). If only one scale is used, the main objects are favoured in the descriptor at the expense of less visually salient objects and the representation of the image is partial.

2.2. Method description

We present two contributions to large scale visual concept detection. The first focuses on training CNNs in absence of manually validated data. This contribution is important because it bypasses the cumbersome image annotation step that is usually performed before visual concept learning. In the context of USEMP, learning from Web data allows one to make possible with little manual effort the domain adaptation for privacy-related concepts, a domain that is poorly covered by existing image datasets. The second contribution relates to an enhancement of the Semfeat descriptor introduced in D5.2 with local information obtained from image regions and with an adaptable sparsification technique. Combined, these two contributions improve the classification performance of the descriptor and thus the image-based profiles of USEMP users.

2.2.1. Effective training of convolutional neural networks with Web images

In D5.2, we introduced Semfeat, a scalable semantic image descriptor built on top of mid-level CNN features. These were learned using a training dataset that was obtained through a manual annotation process that is difficult to scale up. As an alternative, we propose a full learning pipeline that exploits Web images for CNN training instead of a manually built dataset. When collecting images from the Web in order to illustrate visual concepts, a major challenge is related to the noisy character of the obtained dataset. To address this challenge, we designed a bootstrapping pipeline that includes four main steps: (1) image collection from the Web; (2) training with raw Web images; (3) dataset re-ranking; (4) fine tuning with re-ranked images.

Web image collection is done using two data sources, Flickr and Bing, in order to represent each concept with a large number of images. This choice is made since richer sets of images generally lead to better classification performance. In addition to the number of images, the use of two different sources ensures a larger diversity of the visual representations, which is likely to improve the generalization capability of the trained model. To compare our approach with the one based on ImageNet images, we populate the 1,000 synsets that are included in the ILSVRC collection. As a result, we obtain a noisy collection that includes a total of 3.14 million images with 70% coming from Flickr and 30% from Bing.

The second step of the approach pertains to the **training of a classification model** directly with Web images. We first test the performance of the AlexNet architecture introduced in (Krizhevsky et al., 2012) in order to assess the effect of noisy images on classification performance. The results of the ILSVRC competition (Russakovsky et al., 2014) show that, generally, the deeper a network is, the better the obtained results will be. However, the learning process is more complicated for deep networks due to memory limitations but also due to a more difficult convergence process. To account for these limitations, we introduce an architecture called FB (for Flickr-Bing) that includes 13 layers. The network is trained on a GTX Titan-X GPU card; every 1000 iterations take 80 minutes and the process is stopped after 350,000 iterations. Despite the presence of noise, the model converges and can be used as a baseline for further experiments.

The third step of the approach exploits **reranking techniques** to reduce the negative influence of noisy Web images that represent the visual concepts. Three techniques were investigated:

- *Cross-validation (CV)* – splits the Web images associated to a visual concept into K disjoint subsets. A single rejection class that includes images of a variety of other concepts is built. Given a target subset, a binary SVM is built with the remaining (K-1) subsets as positive examples and images of the rejection class as negative examples. A linear SVM is chosen in order to ensure the scalability of the reranking method. In this setting, each target image is classified against the corresponding SVM and the images that are ranked highest are favored in the class representation. CV is an unsupervised reranking method since it does not require any manual annotation of images in order to work.
- *Kernel Mean Matching (KMM)* (Huang et al., 2006) - reweights unlabeled data with respect to a labeled dataset in such a manner that the weighted arithmetic means of the two sets are approximately equal.
- *Transductive Support Vector Machine (TSVM)* (Sindhwani et al., 2006) – assumes that efficient reranking can be achieved with a labeled dataset whose size is much smaller than that of the unlabeled examples that need to be ranked.

Finally, a **fine tuning of the CNN** model is performed by initializing the learning process with the FB model obtained during the second step in order to accelerate the process. More importantly, given that the application of reranking techniques reduces the number of examples per concept, this initialization also helps with reducing the potential effect of data scarcity.

2.2.2. Enhancement of semantic features for image mining

Semantic features (i.e. characterization of an image through a series of semantic concepts such as car, night, city etc.) were introduced in D3.2. There, we showed that the sparsification of such features, which retains only the most informative concepts in the vector associated to the image, is beneficial for image mining. A limitation of the sparsification method introduced in D3.2 is that it retains a fixed number of concepts for all the images, regardless of their actual content. This limitation is illustrated in Figure 2, with two images that have different visual complexity and that should consequently be characterised by a different number of visual concepts. The image to the left being more complex, the proposed "content based sparsity" (CBS) scheme will retain a larger number of concepts compared to a fixed sparsity selection. Concerning the right image, one concept that would have been retained with the fixed scheme is removed by CBS due to its low probability score.

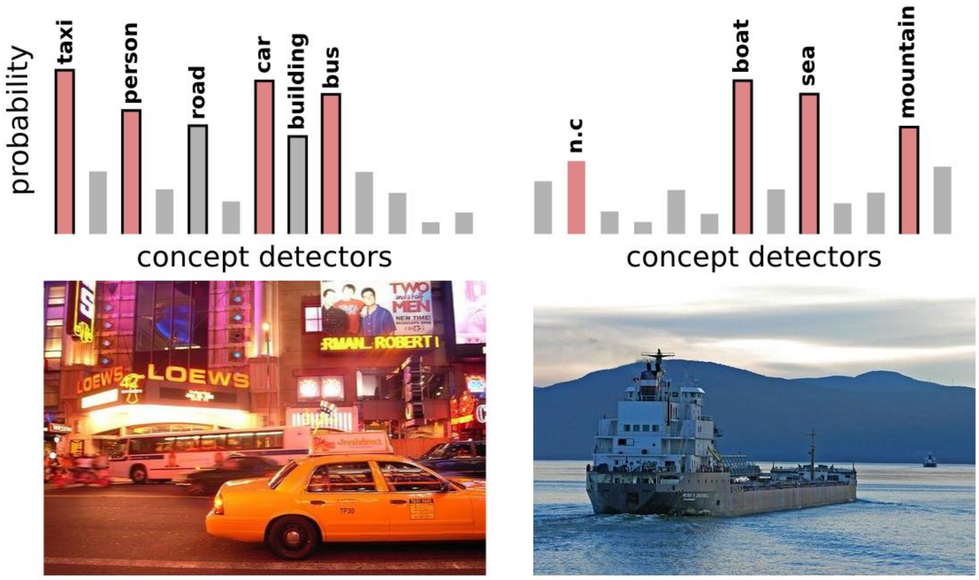


Figure 2. Images with different visual complexity and of associated visual concepts that are retained in adapted semantic features.Using a fixed sparsity scheme with 4 active concepts, the concepts marked in red are retained for each image. Using the proposed adaptation scheme, named "content based sparsity" (CBS), all concepts that are in black boxes are retained. CBS attributes six concepts to the left image, adding two of them to the fixed sparsity scheme, and removes one (n.c.) from the right image due to its low probability.

The number of concepts retained with CBS is computed based on the confidence of concept predictions. We determine the number of active concepts by examining the profile of the non-sparsified semantic signature associated to the tested image. We propose to consider this raw semantic signature as a source of information and to base the sparsification on the Shannon entropy of this source. Put simply, the inverse value of entropy will allow the selection of a small number of concepts for images with low visual complexity, such as the one presented to the right of Figure 2.

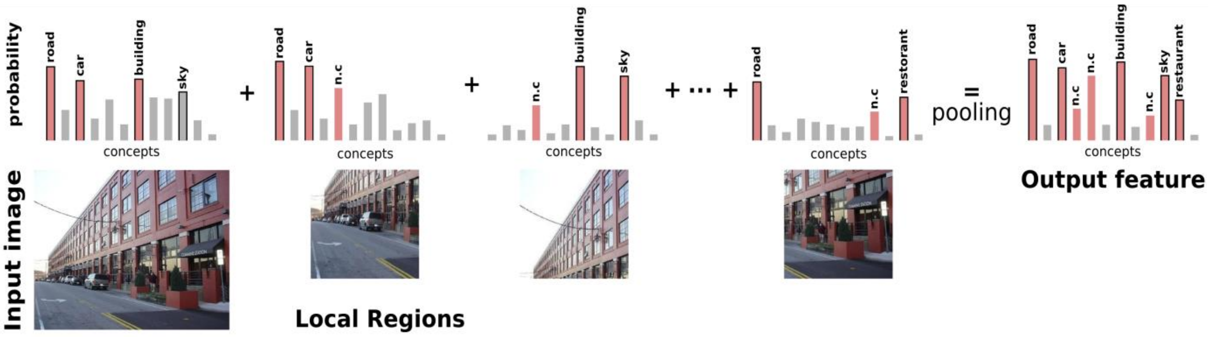


Figure 3. Illustration of the proposed "constrained local enhancement" (CLE) that is applied on top of the CBS selection over the full image and its local region. A max-pooling scheme allows the selection of the most salient concepts that appear either in the full image or in its local regions.

To account for the confidence of the prediction, we use the probability of the top concept among those available in the semantic signature. The CBS selection criterion is expressed in the following equation:

$$s(i) = \propto * \frac{p_{max}(i) * C}{H(i)}$$

where: i is the tested image; $s(i)$ is the sparsification level associated to i; $\propto$ is a normalization parameter that is set through cross-validation; $p_{max}(i)$ is the maximum probability associated to any of the concepts included in the raw semantic signature; C is the total number of concepts available in the raw signature and $H(i)$ is the entropy associated to the image, which is computed over the non-sparsified semantic signature statistic profile.

Another important limitation of the Semfeat feature proposed in D3.2 concerns the fact that concept probabilities are extracted from the entire image and it was shown that choice penalizes small size objects from the image (Razavian et al., 2014). We thus introduce an extension of CBS that also takes into account regions of the image and is named "constrained local enhancement" (CLE). Local image information is added by processing image regions and more precisely the four corners of the images with 2/3 of its full size on each side and the central region of the same size. The computation of CLE, i.e. integration of the local constraints in CBS, is illustrated in Figure 3. It enables an image representation that accounts both for the complexity of the image and for the content that appear in localized regions of it. CLE increases the complexity of the computation since features need to be extracted from each region but the total computation time stays manageable. In its full expression, the computation of CLE takes approximately 500ms and can be further optimized.

2.3. Evaluation and testing

The experimental evaluation of our methods is organized in two main parts, corresponding to the two contributions presented here. In both cases, standard evaluation datasets are used in order to facilitate the comparability of the proposed approaches with other works in the field.

2.3.1. Evaluation of CNN training with Web images

The classification models obtained with Web data are evaluated in a transfer task, i.e. learning on a given set of concepts and testing on datasets whose concepts are at best partially covered in the training dataset. The following evaluation datasets are used in the evaluation:

- VOC07 – dataset that includes 20 diversified concepts embedded in complex settings.
- Flowers102 – specialised dataset that includes 102 flower species.
- MIT67 – 67 indoor scenes.
- Action40 – 40 human actions.
- Caltech256 – 256 objects captured in controlled settings, i.e. objects are centered and in focus in the image.
- SUN 397 – 397 outdoor scenes.

A first experiment is done using the AlexNet architecture. The reference results (AN_{REF}) are obtained with this architecture and manually labelled images from ImageNet. The results with raw Web images are presented as AN_{RAW}. Corresponding to three re-ranking schemes, three versions of fine-tuning are tested for the AlexNet architecture: AN_{CV}, AN_{KMM}, AN_{TSVM}.

The results presented in Table 1 show that the results obtained with Web data are, in most cases, close to those obtained with manually labelled ImageNet images. This result is very important because it shows that CNNs are able to cope with a reasonable amount of noise in the training set, provided that enough data is available for each concept. Collection is done through Web image search engines by launching queries with concept names. If we discard noise, the main difference between the Web and ImageNet data used here is the average number of images per concept, i.e. 3,140 vs. 1,200 respectively. The proposed re-ranking techniques bring a small improvement compared to raw Web data in most cases and further narrow the gap with the manually labelled dataset. While it is difficult to conclude clearly on the superiority of one of the three techniques, AN_{CV} gives the best results on average.

	AN_{REF}	AN_{RAW}	AN_{CV}	AN_{KMM}	AN_{TSVM}
VOC07	71.7	70.8	71	70.6	70.4
Flowers102	87	87.6	88.4	88.6	89.4
MIT67	56	52.5	53.1	53.7	51.9
Action40	60.2	55.9	56.5	56.1	56.3
Caltech256	70.3	68.6	69.5	68.9	69.7
SUN397	46.1	45.8	46.1	45.4	45.4

Table 1. Classification results on different standard datasets using the AlexNet architecture with manually labeled and Web data respectively. The evaluation metric is the Mean Average Precision (mAP) score.

A second experiment compares CNN architectures of different depths, including the AlexNet architecture with ImageNet and Web data (AN_{REF} and AN_{RAW}), the FB architecture with 13 layers presented above and VGG_{16}, one of the best performing systems at ILSVRC 2014 (Russakovsky et al., 2014).

	AN_{REF}	VGG_{16}	AN_{RAW}	FB_{RAW}
VOC07	71.7	79.9	70.8	76.6
Flowers102	87	87.5	87.6	88.8
MIT67	56	67.1	52.5	61.6
Action40	60.2	72.6	55.9	63.3
Caltech256	70.3	77.9	68.6	75.2
SUN397	46.1	53.8	45.8	51

Table 2. Classification results with CNN architectures of different depths. The evaluation metric is the Mean Average Precision (mAP) score.

The results presented in Table 2 confirm the importance of CNN architecture depth, since $VGG_{16,}$ the best performing configuration, is also the deepest one. FB_{RAW}, trained with raw Web data, has the second best performance and outperforms the AlexNet architecture trained with both noisy and labelled data.

2.3.2. Evaluation of the enhancement of semantic features

The proposed "constrained local enhancement" (CLE) scheme, that incorporates image-adapted sparsification and local enhancement, is evaluated in image classification and content-based image retrieval tasks. The following datasets are used:

- VOC07 – dataset that includes 20 diversified concepts embedded in complex settings. It is used in classification and retrieval experiments
- VOC12 – extension of the VOC07 dataset, with more training and test images. This dataset is used only for classification experiments.
- MIT67 – dataset that contains 67 indoor scenes. This dataset is used only for classification experiments.

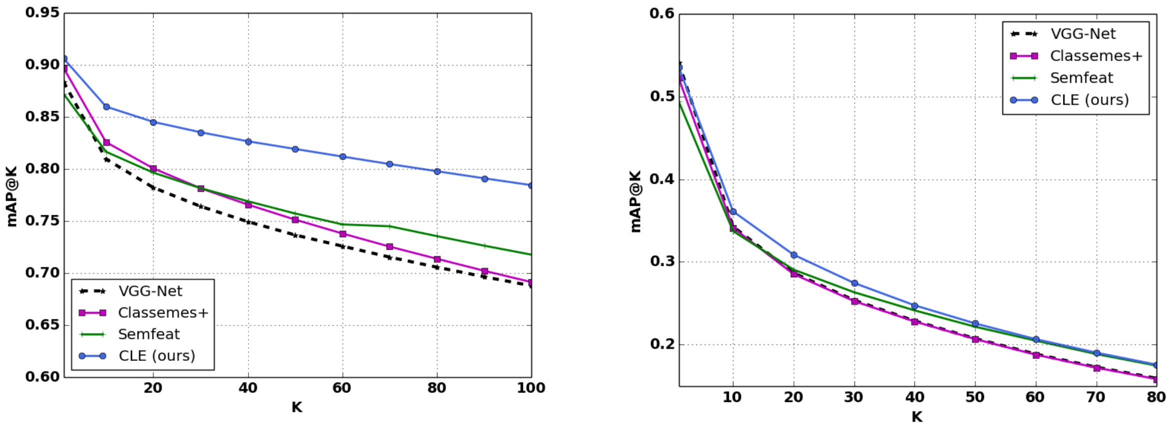
In all experiments, the visual concept detection is performed with linear SVMs that are learned on top of the fc7 layer of the VGG features proposed by (Simonyan and Zisserman, 2014). The VGG features are used as a strong baseline in our experiments, along with the following other approaches:

- (Oquab et al., 2013) – mid-level features that are learned with a limited amount of training data after transfer from a CNN model learned with massive data.
- Classemes+ – our own implementation of classemes features described in (Bergamo et Torresani, 2012). The main differences come from (1) the fact that these semantic features are built on top of more powerful VGG features instead of a host of handcrafted low-level features and (2) from the fact that we exploit linear SVMs instead of approximation of a non-linear classifier.
- Semfeat – sparsified version of Classemes+, with a fixed sparsity scheme. The sparsification factor is set at 50 for all three bases since this value gives an optimal result.

	(Oquab et al., 2013)	VGG	Classemes+	Semfeat	CLE
VOC07	77.7	86.1	82.4	82.8	88.2
VOC12	-	84.5	81.7	81.7	86.6
MIT67	69.0	48.7	58.9	61.5	71.6

Table 3. Classification results with different visual features on three standard datasets.The results are reported with the standard evaluation measures for the datasets, i.e. mAP for VOC07 and VOC12 and classification accuracy for MIT67.

The results presented in Table 3 show that CLE outperforms all other tested approached on the three evaluation datasets. Notably, CLE has better results than VGG, the mid-level CNN feature upon which it is based. CLE is also significantly better than Classemes+ and Semfeat, two competitive semantic features that were proposed in literature. The adaptation of features to the visual complexity of the image and the addition of a locality constraint proved to be beneficial and allowed CLE to be the only semantic feature that outperforms the strong CNN feature used for their implementation.



(a) (b)

Figure 4. Retrieval results with CLE and the top three performing baseline features: VGG, Classemes+ and Semfeat. Performance is measured using mAP at different recall levels. Figure (a) shows results on the PascalVOC07 collection, while Figure (b) illustrates the results on MIT Indoor 67.

Retrieval experiments are run with VOC07 for object retrieval and MIT67 for scene retrieval. Only the best three baselines from classification experiments are reused here: VGG, Classemes+ and Semfeat. We use the standard training and test sets as collection for both datasets. The results are presented in Figure 4 and show that CLE outperforms all baselines for both datasets and at all recall levels. The difference is higher for object retrieval (Figure 4a), where an improvement of up to 10% is obtained compared to VGG.

2.4. Implementation and usage

The implementation of the concept detection is similar to the one presented for the tool version introduced in D5.2. In particular, the interface of the tool is the same in order to minimize the effect of changes on the integration process.

2.5. Next steps

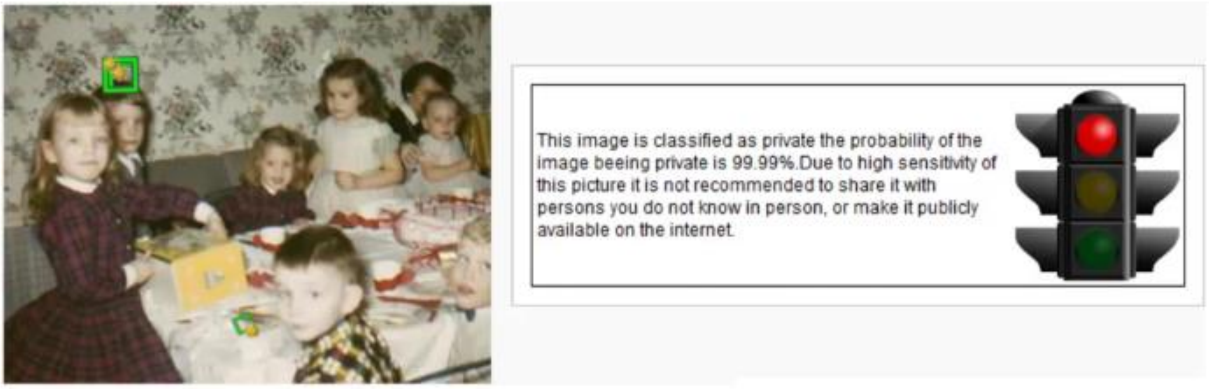
In the remaining months of USEMP, focus will be put on assisting with any problems arising from the integration of the tool and in pushing all concepts detection updates in Databait before the end of the project.

3.Private/non-private image classification

Uploading and sharing images in Online Social Networks (OSNs) is nowadays a commonplace activity for the majority of Internet users. Image sharing is so pervasive and frequent that many people become gradually oblivious to the fact that the content of the images they share is visible by several other members of the OSN and are accessible by the OSN itself. Typically, OSNs employ sophisticated algorithms to make sense of the data posted by their users in order to create personal profiles that they often use to perform ad targeting. Although image sharing often has no direct consequences for the individual sharing them (other than a sense of reward when peer OSN members approve of them), there are often implications that cannot be foreseen or realised at the time of sharing. For instance, implicitly disclosing one's location through their images (e.g., by making it obvious that they are away from home, on holidays) could be maliciously used to compromise the security and privacy of individuals (Friedland & Sommer, 2010). As another example, consider the implicit disclosure of one’s smoking or drinking habits being used by an insurance company to adjust (increase) the insurance cost. Furthermore, the implications are not limited to the uploader but extend to other people depicted and/or “tagged” in the image. (Minkus et al., 2015), for instance, studied how parents often compromise the privacy of their own children by posting information about them online.

Clearly, since image sharing is such a widely used and valued service, preventing OSN users from sharing their images cannot be considered as a viable means of protecting their online privacy. Instead, having access to a service that could automatically process one's images before they are shared with the OSN, and being alerted in case their content is found to be sensitive, would be a very practical and transparent way of safeguarding the online privacy of OSN users without affecting their image sharing experience.

A first solution was presented in (Zerr et al, 2012), where the authors considered the problem of automatically classifying users' images as being of private or public nature, and tested the effectiveness of standard image and text features in a supervised learning setting for solving the problem. In that work, the authors focused on developing models that capture a generic (“community”) notion of privacy, making the underlying assumption that each user perceives privacy in the same way. However, OSN users oftten have wildly different perceptions and norms regarding privacy (Paine et al., 2007). A further limitation of that solution is that the classification decision was justified by highlighting the most dicriminative local patches in the image as shown in Figure 5. Such a justification is hardly comprehensible for non-experts in computer vision. Providing more intuitive, higher-level, justifications of the classifier’s decisions would be clearly more desirable.



This image is classified as private the probability of the image beeing private is 99.99%.Due to high sensitivity of this picture it is not recommended to share it with persons you do not know in person, or make it publicly available on the internet.

Figure 5: The justification provided for a private classification by the PicAlert system.

In D5.5, we worked towards developing a personalised image privacy classification system that provides an effective privacy safeguarding mechanism on top of image sharing OSN facilities, while at the same time it alleviates the limitations of previous solutions. In particular, we make the following main contributions:

- Real-world dataset: We create a new realistic benchmark dataset via a preliminary study where users annotate their own photos into private/public according to their personal notion of privacy. Experiments on this dataset reveal the limitations of adopting a generic privacy definition and the value of personal privacy classification models (Subsection 3.3.2).
- Personalised privacy classification: We demonstrate that by combining feedback from multiple users with a limited amount of user-specific feedback, we can obtain significantly more accurate privacy classifications compared to those obtained from a generic model (Subsection 3.3.3).
- Semantic justification: By employing the semfeat representation (introduced in D5.2), i.e. a new type of semantic features, we manage to provide comprehensible explanations of privacy classifications and discover valuable insights with respect to users' privacy concerns. Importantly, these features are computed based solely on the visual content of the images and, therefore, the approach does not require the existence of hand-given image tags.
- State-of-the-art performance: By using visual features extracted from deep convolutional neural networks (CNNs) we significantly improve the state-of-the-art performance on an existing private image classification benchmark (Subsection 3.3.2).

3.1. Related work

In the work of (Zerr et al, 2012), a large-scale user study was conducted that asked participants to annotate a large number of publicly available photos from Flickr as being either “private” or “public”. The study was set up as a social annotation game where players were instructed to adopt a common definition of privacy: “Private are photos which have to do with the private sphere (like self-portraits, family, friends, your home) or contain objects that you would not share with the entire world (like a private email). The rest is public." and were rewarded for providing similar annotations to other players. The resulting dataset, referred to as *PicAlert*[3], was used to train supervised classification models that capture a generic (“community”) notion of privacy.

Extending that work, (Squicciarini et al., 2014) identified more discriminative visual and metadata-derived features and achieved better prediction accuracy. Moreover, (Squicciarini et al., 2014) attempted to solve a more complex privacy classification problem where three degrees of disclosure (view, comment, download) were jointly classified into five privacy levels (“Only You”, “Family”, “Friends”, “SocialNetwork”, “Everyone”). Similarly, to (Zerr et al, 2012), the resulting models capture a generic perception of privacy.

In a recent work (You et al., 2015), the authors deal with the inference of OSN users' attributes and interests using images posted to Pinterest. It was shown that by analysing the content of the images and leveraging their groupings (manually defined by users) into pin boards, user interest profiles can be created. In our work, we propose a method for creating user privacy profiles by combining user feedback on image privacy with the outputs of a large-scale concept detector.

3.2. Method description

3.2.1. *YourAlert*: A Realistic Private Image Classification Benchmark Dataset

Despite its merits, there are two limitations that make *PicAlert* unsuitable as a realistic image privacy classification benchmark: a) it consists of publicly available images with few of them being of really private nature, b) the ground truth collection process makes the unrealistic assumption that all OSN users have common privacy preferences. As a result, a privacy classification model trained on this dataset may practically fail to provide accurate classifications (as shown in Subsection 3.3.2). Moreover, the variability of privacy preferences among users is not taken into account when evaluating the accuracy of privacy classification methods on *PicAlert*, resulting to overly optimistic performance estimates.

To overcome these limitations, we created a new privacy-oriented image dataset with two goals: a) the development of personalised image privacy models, and b) the realistic evaluation of both generic and personal image privacy models. To this end, we conducted a realistic user study where we asked users to provide privacy annotations for photos of their personal collections. To reduce the concerns associated with sharing personal images, we provided users with software that automatically extracts several types of visual features (described in Subsection 3.3.1) from their images and asked them to share the features and the corresponding annotations (instead of the original images). To let participants of the study freely adopt their own notion of privacy, we vaguely described as public “images that they would share with *all* OSN friends or even make them publicly visible” and as private “images that they would only share with *close* OSN friends or not share at all”. To ensure a sufficient coverage of both classes we asked each participant to provide at least 10 private and 30 public images. The current version of the dataset[4] (features and privacy annotations), named *YourAlert*, contains 184 private and 400 public photos contributed by 10 different participants (mainly employees at CERTH and CEA).

[3] Publicly available at http://l3s.de/picalert

[4] The user-study is still ongoing; we expect that the dataset will expand further upon its completion.

3.2.2. Personalized Private Image Classification Models

Privacy classifications based on a generic privacy classification model as the one developed in (Zerr et al, 2012) are undoubtedly useful for preventing users from uploading images that are considered to be private according to a generic notion of privacy. However, as the perception of privacy varies greatly among users depending on factors such as age, social status and culture, it is expected that a generic model would provide inaccurate predictions for certain users, thus decreasing the reliability and usefulness of the alerting mechanism. To overcome this issue, we propose the exploitation of user feedback in order to build personal privacy models.

Given a sufficient amount of user feedback, a personal privacy model could be learned based only on user-specific training examples. However, this requires considerable effort from the user and cannot be guaranteed. Therefore, we propose the construction of an additional type of (semi-)personal models where user-specific training examples are combined with examples provided by other users. In this case, user-specific examples are assigned a higher weight in order to increase their influence on the resulting model. Despite being a seemingly counter-intuitive choice, we show that models trained on such mixtures of examples outperform models that use only user-specific examples when a limited amount of user feedback is available (Subsection 3.3.3).

3.2.3. Visual and Semantic Features

In our experiments we focus on privacy estimation based only on the visual content of the images and extract the following types of state-of-the-art visual features from *PicAlert* and the newly composed *YourAlert* dataset:

- *vlad*: d=24,576-dimensional VLAD+CSURF vectors (Spyromitros-Xioufis et al., 2014) are extracted using a 128-dimensional visual vocabulary and then projected to d'=512 dimensions with PCA and whitening.
- *cnn*: were described as part of D5.2 and are standard convolutional neural network features using the Caffe reference model (Jia, 2013), which is a slightly modified version of the one in (Krizhevsky et al., 2012). We use the output of the last fully connected layer (fc7), which consists of 4096 dimensions.
- *semfeat*: were introduced in D5.2 and are semantic image features obtained by exploiting the outputs of a large array of classifiers, learned with low-level features (Bergamo & Torresani, 2012). Here, we compute the *semfeat* descriptor based on 17,462 ImageNet concepts, as proposed in (Ginsca et al., 2015). In particular, concept models are learned independently as binary classifiers, with a ratio of 1:100 between positive and negative examples. The resulting features are sparsified in order to retain only the top n = 100 classifier outputs for each image. Compared to *vlad* and *cnn*, *semfeat* has an important advantage in our use case, since it enables result explainability: users can obtain human-understandable feedback about why an image was classified as private or not, in the form of top concepts associated to it (which are at the same time important for the privacy classification model).

3.3. Evaluation and testing

3.3.1. Experimental Setup

Throughout the experiments, we use the LibLinear (Fan et al., 2008) implementation of L2-regularised logistic regression as the classification algorithm as it provided a good trade-off between efficiency and accuracy compared to other state-of-the-art classifiers in preliminary experiments. Moreover, the coefficients of a regularised logistic regression model are suitable for identifying features that are strongly correlated with the class variable (Hastie et al., 2001), thus facilitating explanation of the privacy estimates when features with a semantic interpretation such as *semfeat* are used.

To evaluate the accuracy of each privacy classification model we use the area under the ROC curve (AUC). This was preferred over other evaluation measures due to the fact that it is unaffected by class imbalance and it is independent of the threshold applied to the probability outputs of a logistic regression model in order to transform them into hard 1/0 (private/public) decisions. Moreover, AUC has an intuitive interpretation: it is equal to the probability that the classification model will assign a higher privacy score (probability) to a randomly chosen private image than a randomly chosen public image. Thus, a random classifier has an expected AUC score of 0.5 while a perfect classifier has an AUC score of 1.

3.3.2. Generic Image Privacy Classification Models

This section evaluates the performance of generic privacy estimation models under two different settings: a) one where the images to be classified are annotated according to a generic definition of image privacy, b) a more realistic setting where each image is annotated by a different user according to his/her personal notion of privacy. In the first case, the evaluation setting coincides with the one adopted in (Zerr et al., 2012), i.e. the privacy classification models are trained on a randomly chosen 60% subset of the *PicAlert* dataset and tested on the remaining 40%. In the second case, the privacy estimation models are trained on the same subset of *PicAlert* as above, but the testing is carried out on the examples of the *YourAlert* dataset. Figure 6 shows the AUC scores obtained on *PicAlert* and *YourAlert* when the state-of-the-art visual features described in Subsection 3.2.3 are used. On PicAlert, we also show the performance of the two best performing visual features used in (Zerr et al., 2012): a) quantized SIFT (*bow*) and b) edge-direction coherence (*edch*) vectors.

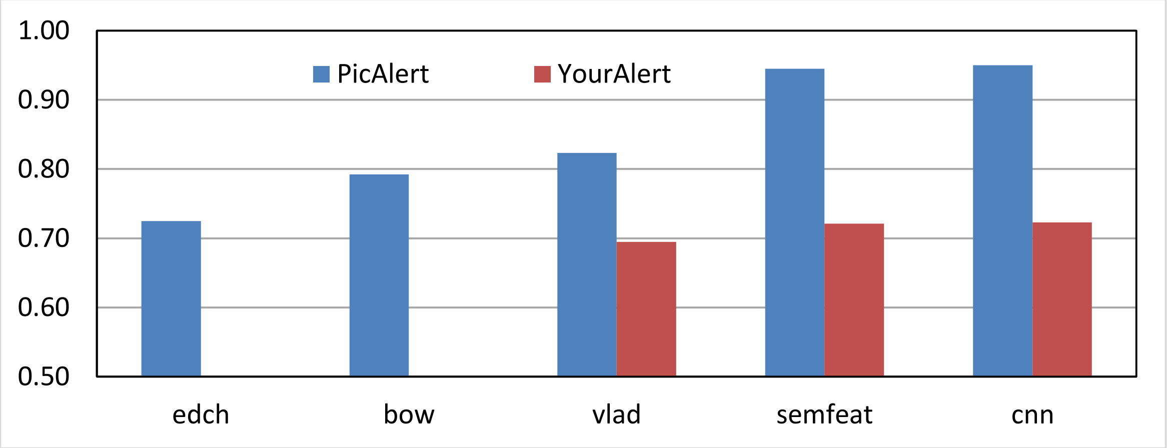


Figure 6: AUC performance of generic image privacy classification models on PicAlert and YourAlert

Looking at the performance on *PicAlert*, we see that *vlad*, *semfeat* and *cnn* lead to significantly better results than *edch* and *bow*. With *cnn*, in particular, we obtain an AUC close to 0.95 which is 20% better than the AUC score obtained with *bow* (the best visual feature of those used in (Zerr et al., 2012)). *semfeat* have very similar performance with *cnn*, a fact that makes them a very appealing choice, given their sparsity and interpretability properties.

Despite the impressive results obtained by the generic privacy models on *PicAlert*, we see that their performance drops dramatically on *YourAlert*. To confirm that this drop in performance is not due to a lack of training examples, we trained generic privacy models using an increasing number of generic training examples from *PicAlert*. As show in Figure 7, using additional generic examples does not help in attaining better performance on *YourAlert*. After a sharp increase from 100 to 1000 training examples, AUC performance reaches a plateau and does not change significantly after 5000 examples with all types of features. Figure 8 presents a breakdown of the generic model's performance on each user of the YourAlert dataset, i.e. a separate AUC score is calculated for each user, with *vlad*, *semfeat* and *cnn* features. In all cases we see a large variability in performance across users ranging from nearly perfect (u4) to almost random (u8 and u9). These results are in accordance with our hypothesis and highlight the necessity of developing personal privacy classification models.

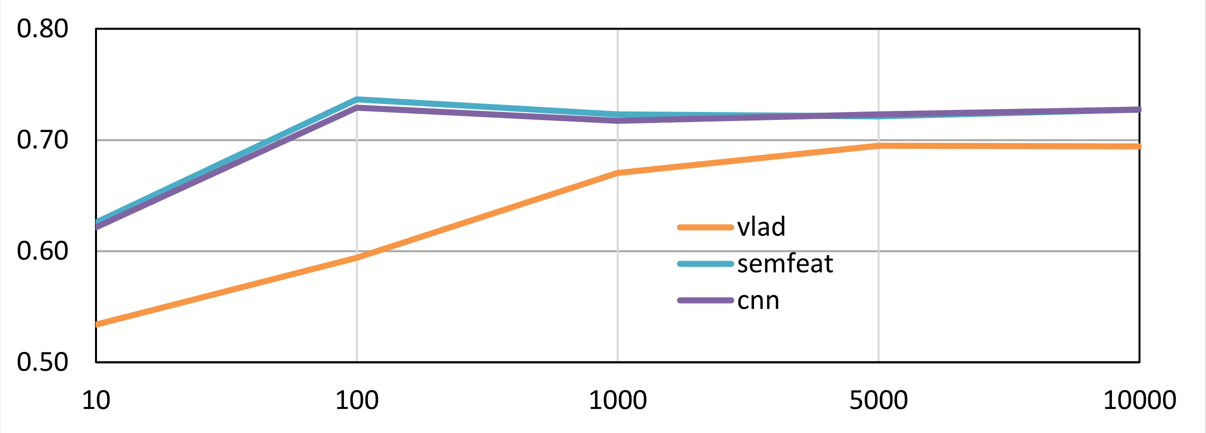


Figure 7: AUC performance of generic models on YourAlert as a function of the number of examples.

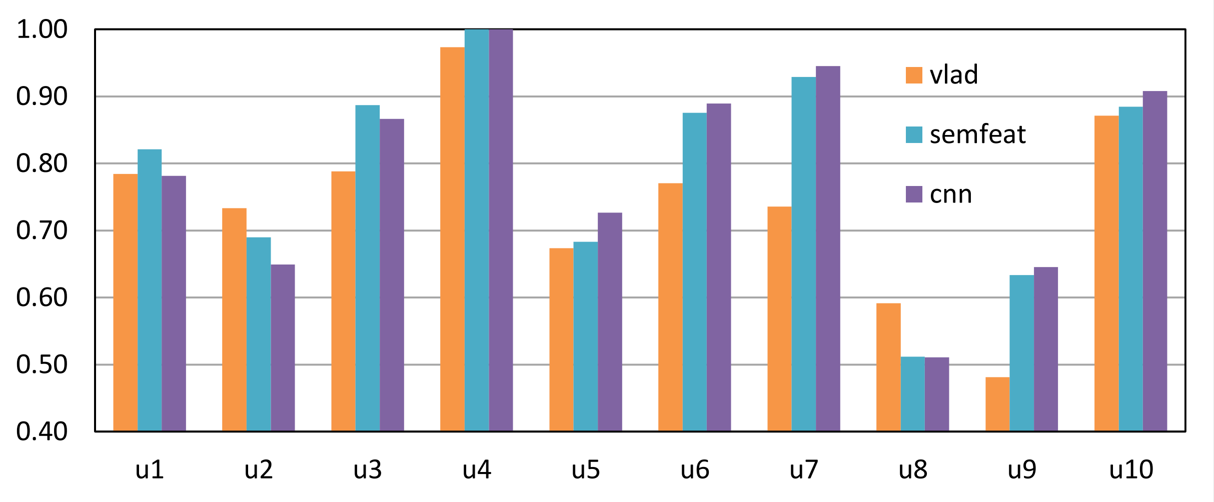


Figure 8 : AUC performance of generic models on the images of each user of the YourAlert dataset.

3.3.3. Personalized Privacy Classification Models

This section compares the performance of generic privacy classification models with the performance of personalized privacy classification models that exploit user feedback in order to adapt to specific users. In particular, two types of personalised models are constructed and evaluated on *YourAlert*:

- *user*: A different model is built for each user using a subset of the YourAlert examples that has been annotated by that particular user.
- hybrid*:* A different model is built for each user that combines a subset of the YourAlert examples that has been annotated by that particular user with examples that belong to other users. As discussed in Subsection 3.2.2, user-specific examples are assigned a higher weight in order to have a greater influence on the resulting model compared to the rest of the examples.

The evaluation of the personalised models is carried out using a modified k-fold cross-validation procedure which ensures that unbiased, out-of-sample estimates are obtained for all examples of each user. In particular, the examples contributed by each user are randomly partitioned into k folds of equal size. Out of these, a single fold is retained as the validation set and used to test the model, and from the remaining k-1 folds we randomly select a specified number of examples and use them as training data either alone (user models) or together with examples from other users (hybrid models). This process is repeated k times, with each of the k subsets used exactly once as the validation set. All predictions concerning each user are then gathered into a single bag to calculate a per-user AUC score, or predictions for all users are combined together to calculate an overall AUC score for the examples of the *YourAlert* dataset.

Figure 9 shows the AUC scores obtained on *YourAlert* by *user* and *hybrid* models trained on {5,10,15,20,25,30} user-specific examples when *semfeat* features are used (similar results are obtained with the other types of features). Three variants of the *hybrid* models are constructed, each one using a different weight (w={1,10,100}) for the user-specific examples[5]. The figure also shows the performance of the *generic* model and a model (*other*) that is trained using only examples from other users to predict the privacy of the images of each user.

We first observe that *other* is only slightly worse than *generic*, which suggests that *YourAlert* is sufficient for creating a generic privacy classifier despite its smaller size and the fact that each annotator adopts a personal definition of privacy. Looking at *user* we see that its performance increases fast with more training examples and obtains a similar performance with the *generic* model with only 30 training examples. On the other hand, all *hybrid* models, obtain better performance than the *generic* model even with a very small number of user-specific examples are used. Moreover, we see that the performance of *hybrid* models increases with larger values of w and the best results are obtained by *hybrid w=100* (the performance stops improving with larger values). These results clearly indicate that at the presence of a limited amount of user feedback, combining user-specific with generic examples can lead to better privacy classifications.

[5] To increase the weight of an example in a logistic regression model we simply repeat the example multiple times.

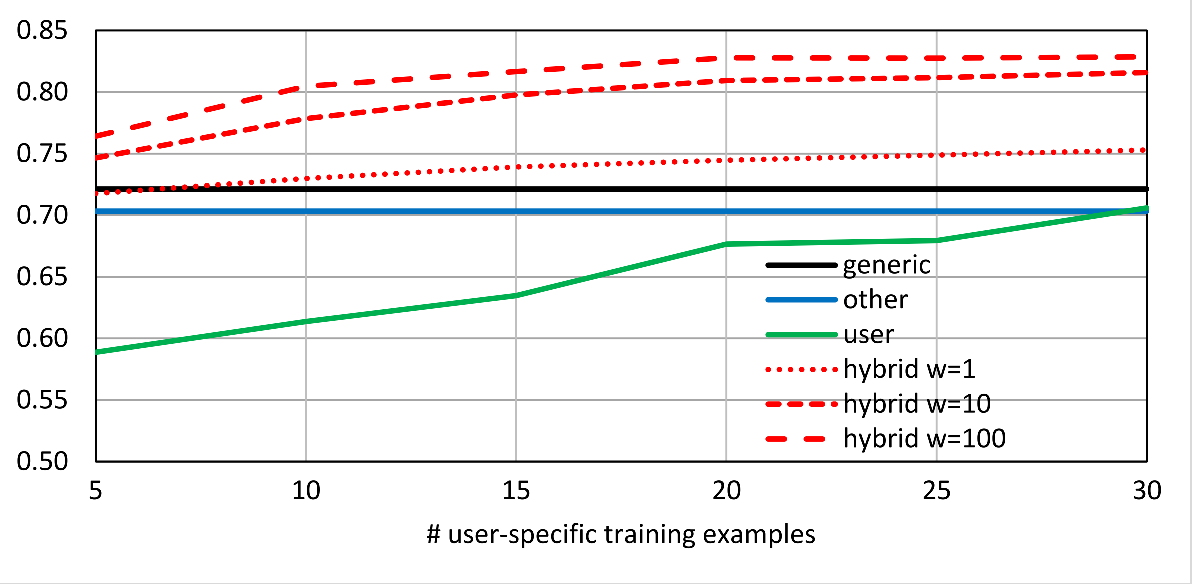


Figure 9: AUC performances of personal models.

3.3.4. Image Privacy Exploration via SemFeat

The use of *semfeat* features can help gain insights into privacy perceptions. Figure 10 shows tag-clouds of the semfeat concepts that are assigned the largest positive (associated with the private class - left) and negative (associated with the public class - right) classification coefficients in the generic model trained on PicAlert. Figure 11 shows a PicAlert image classified as private by the generic model along with a tag-cloud of its most prevalent *semfeat* concepts that can serve as an easily interpretable justification of the classifier's decision.

Moreover, *semfeat* features can help us identify users whose privacy concerns deviate strongly from the average perception of privacy. To this end, we built a single generic privacy classification model using the whole *YourAlert* dataset as well as 10 personalized privacy classification models trained using only the examples contributed by each user. For each model, we identify the features that are assigned the 50 largest positive (associated with the private class) and negative (associated with the public class) classification coefficients and search for features that are strongly correlated to privacy according to the generic model and negatively correlated to privacy according to a personalized model (and vice versa). Despite the fact that less than 1% of the *semfeat* concepts are considered in these comparisons, we can still gain valuable insights. For instance, according to the generic model, concepts related to family and relatives, such as *dad*, *grandfather* and *firstborn* are highly correlated to private images, while concepts related to natural scenes, such as *waterfront*, *snow* and *hillside* are correlated to public images. In addition, we found some interesting deviations from the generic model, e.g. *drinker* is strongly correlated with privacy according to the generic model while it is negatively correlated with privacy for user *u5*. On the other hand, concepts *shore* and *seaside* are private for user *u9* and public according to the generic model.

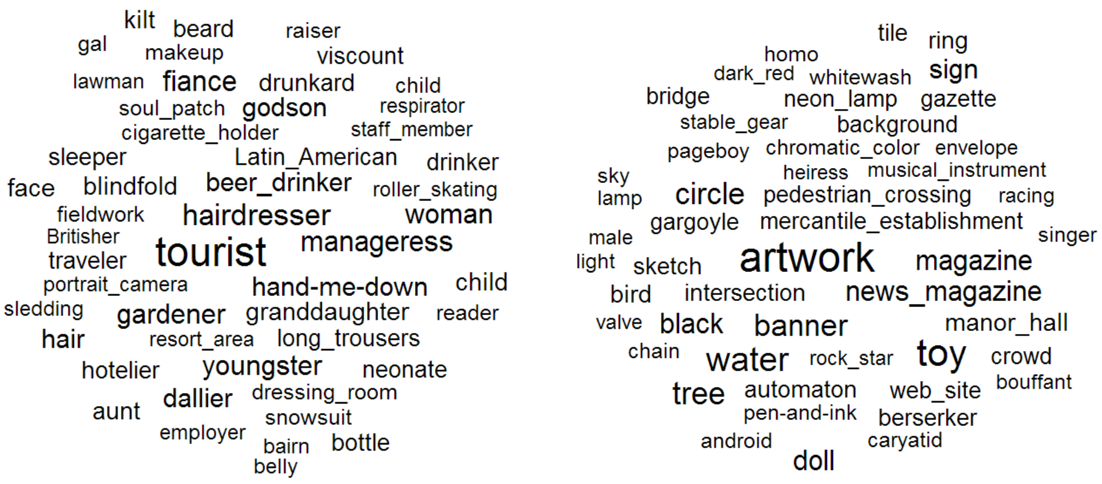


Figure 10 : semfeat concepts associated with private (left) and public (right) images

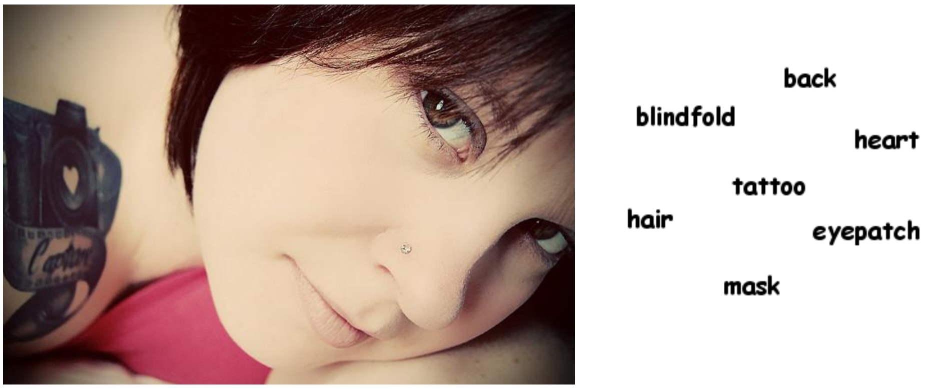


Figure 11: A PicAlert image classified as private along with its most prevalent semfeat concepts.

3.4. Implementation and usage

In terms of implementation we provide an executable jar file (privacy.jar) that can be used to build a generic as well as personalized privacy classification models on the *YourAlert* dataset (with *semfeat* features) and output a csv file that contains the top-k largest positive (associated with the private class) and negative (associated with the public class) classification coefficients for each model. Optionally a tag-cloud can be generated (using the Kumo[6] library) for each set of concepts. To perform these operations, use the following command:

java –jar privacy.jar [dataset] [output-folder] [topK] [tag-cloud]
where
[dataset] full path to the YourAlert dataset (with semfeat features)
[output-folder] full path to the output folder (where the csv and tag-cloud files will be written)
[topk] how many top private and public concepts to be considered (e.g. “50”)
[tag-cloud] whether to generate tag-clouds as well (“true”/”false”)

[6] https://github.com/kennycason/kumo

3.5. Next steps

We presented a framework for privacy-aware classification of personal images. Our main contribution was the demonstration (via experiments on a newly introduced image privacy dataset) that generic privacy classification models may perform very poorly in a realistic setting and that personalization of the models based on user-specific feedback is important to achieve better performance. Furthermore, we experimented with different strategies of utilizing feedback and found that a combination of user-specific with generic feedback yields better performance when only a small amount of user-specific feedback is available. Finally, we introduced a new type of semantic features that led to impressive performance and allowed the discovery of interesting insights regarding the privacy notions of individuals.

In the future, we aim to integrate the developed and tuned private image classification models into DataBait, including appropriate extensions to the User Interface and the user experience. In addition, we aim to expand the YourAlert dataset in terms of both number of participants and in terms of number of contributed photos per participant. This will allow us to draw safer conclusions about the performance of personalised versus generic privacy estimation models and to see what happens when more feedback is available from each user. Moreover, we would like to develop models able to classify a user's photos into a finer number of privacy classes, each one corresponding to a different audience a user would be willing to share a photo with (e.g. 'close friends', 'acquaintances, 'all friends', 'friends of friends', 'public'). Finally, with respect to the semantic interpretation of privacy classifications we would like to construct a new vocabulary for *semfeat* features that is more focused into privacy-related concepts.

4.Logo detection

Logo and product recognition are useful in order to create consumer profiles for users, one of the core privacy dimensions determined in WP4 and WP6. Through automatic recognition, a profile will include specific brands and products that are likely to be of interest for a particular user. Visual mining can then be combined with the results of text mining (named entity recognition to be integrated in T5.1) and with user likes (analysed in T6.1) to obtain a more complete profile. Logo and product recognition is mainly useful for the first use case of the project as it contributes to determining the value of the user's personal data. The first experiments carried out for logo recognition were performed through an adaptation of an existing method and results were reported in D5.2.

4.1. Related work

Many computer vision tasks, including object classification, localisation, as well as content based retrieval, are increasingly tackled with convolutional neural network (CNN) architectures (Russakovsky et al., 2014, Razavian et al., 2014). CNNs globally provided a leap in performance for these tasks. CNNs are successfully used to recognise a wide variety of categories: birds (Branson et al., 2014), flowers (Razavian et al., 2014), cars (Yang et al., 2015), traffic signs (Sermanet et al., 2011), human faces (Sun et al., 2014), indoor/outdoor scenes (Zhou et al., 2014), etc. Following these progresses and to improve our system, we decided to use CNN architectures for logo recognition.

4.2. Method description

Based on the results reported for concept detection using a Web corpus, we take a similar approach for logo detection. We trained a 16-layer deep network described by (Simonyan et Zisserman, 2014) to recognize up to a total of 1,467 different logos and products. We initialise the network with weights from a precomputed model on the ImageNet dataset. We chose this over a random initialisation because, in Girshick et al. (2014), it is suggested that given limited and domain-specific data, fine-tuning a pre-trained CNN model can be an effective and practical approach. In other words, it enables us to train a model with limited data. We apply data augmentation on images, i.e., rotations, colour jittering, perspective change, zooms, during the training process. It has been shown that data augmentation strengthens the robustness of a model (Wu et al. 2015). To extract the features from an image, we feed it to the network which produces a single compact vector representation. The L2-distance is then used to search similar images within the base.

4.3. Evaluation and testing

We conducted an experiment to evaluate the method using FlickrLogos-32, a publicly available dataset[7], which facilitates comparison between our approach and the state of the art. The dataset contains photos showing brand logos and is meant for the evaluation of logo retrieval and multi-class logo detection/recognition systems on real-world images. We collected logos of 32 different logo brands by downloading them from Flickr. All logos have an approximately planar surface. The retrieved images were inspected manually to ensure

[7] Available at http://www.multimedia-computing.de/flickrlogos (accessed on 23/12/2014)

that the specific logo is actually shown. The whole dataset is split into three disjoint subsets, each containing images of all 32 classes. The training set consists of 10 images that were hand-picked such that these consistently show a single logo under various views with as little background clutter as possible. The other two partitions Pv (validation set) and Pt (test set = query set) contain 30 images per class. Unlike the training set, these images contain at least one instance of a logo but in several cases multiple instances. Both partitions Pv, and Pt include another 3000 images downloaded from Flickr with the queries "building", "nature", "people" and "friends". These images are the negative images and complete the dataset.

We evaluated our method using a *retrieval* evaluation approach:

- Images from the training and validation set are indexed, including non-logo ones (4280 images)
- The 960 images of the query set (logos) are used as queries.
- The mean average precision (MAP) is used to measure the detection accuracy.

Our CNN method resulted into a MAP of 0.88 while our previous method reported in D5.2 had a MAP of 0.48. We are now better by a high margin over the best reported result in the literature on this benchmark (0.55, Romberg et al., 2011).

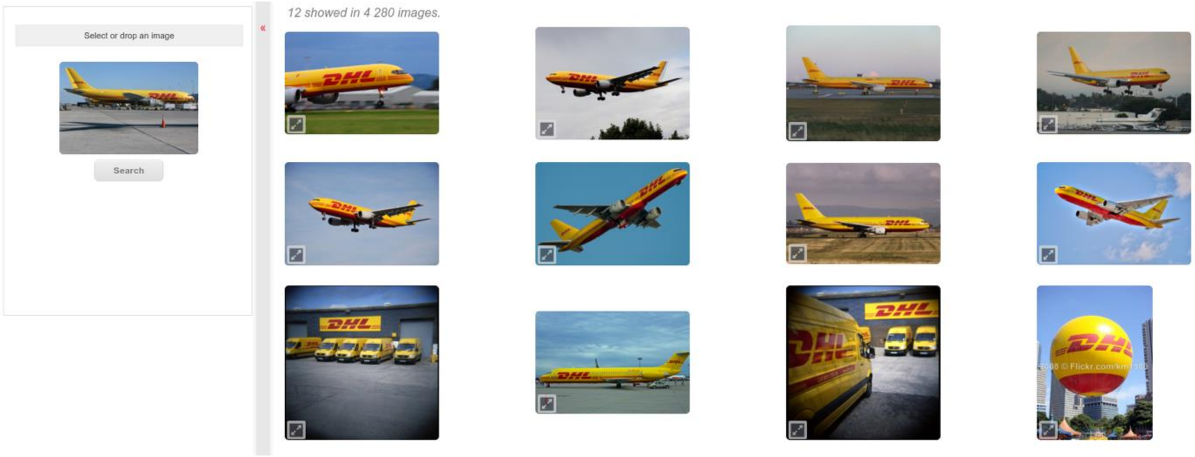


Figure 12. Illustration of logo detection process. The query image is to the left of the figure and its nearest neighbours to the right.

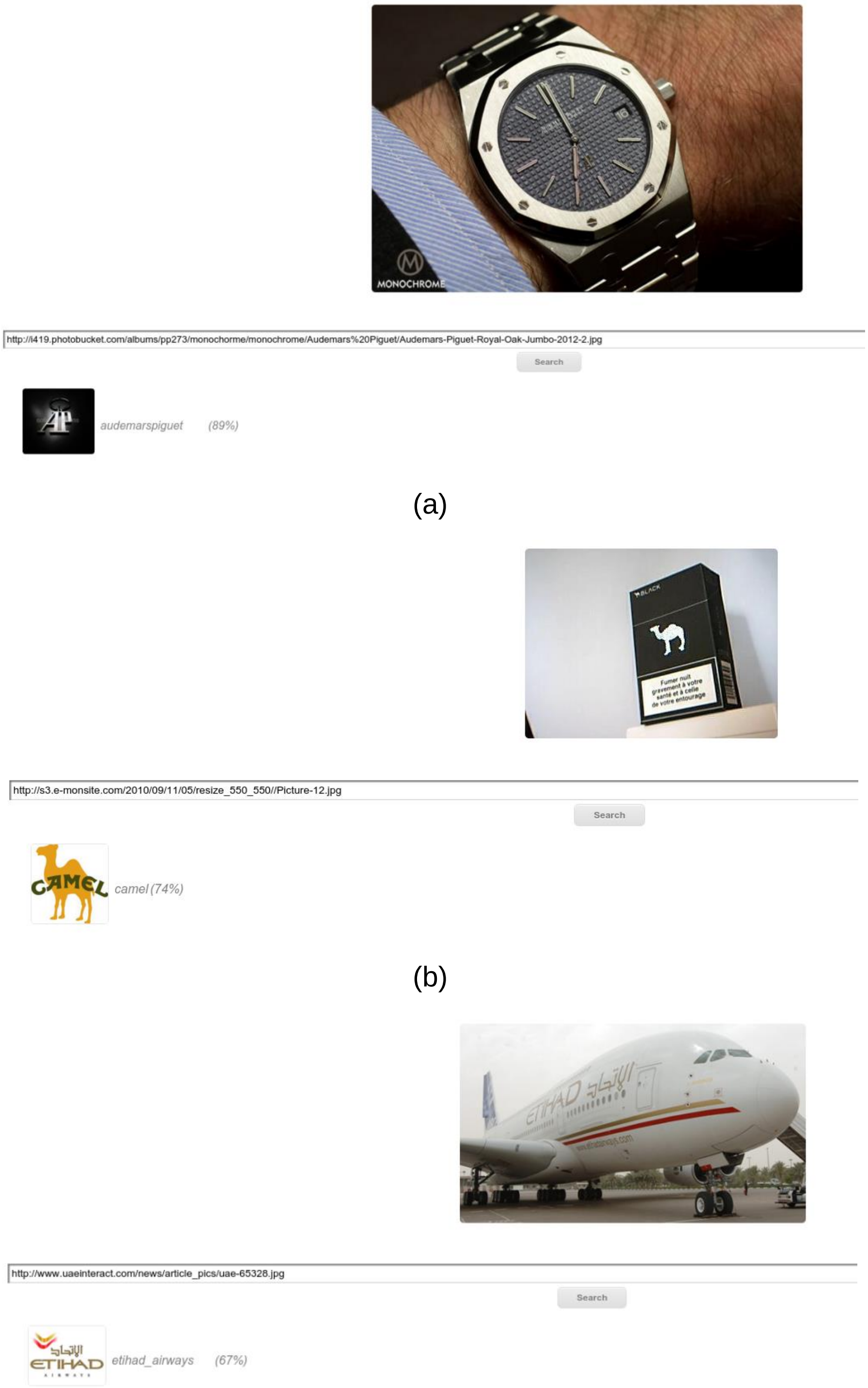
We present an illustration of logo detection process in Figure 12. Illustration of logo detection process. The query image is to the left of the figure and its nearest neighbours to the right. The query image (on the left) is a plane in the colours of DHL with two logos on it. Note that the logo itself does not cover a wide part of the image. The same logo appears in similar images, mostly planes. We can also see that more complex images are returned. A qualitative examination of logo detection results shows that, as expected, the method fails when the logo is too small or partially occluded.

We also evaluated our method using a *classification* evaluation approach:

- 3,960 images have to be classified;
- 960 images contain logos;
- 3,000 are logo-free;
- The precision and recall are used to measure the recognition accuracy.

Our method reaches 0.993 in precision and 0.86 in recall where the best reported result in the literature is 0.999 in precision and 0.83 in recall (Romberg et al., 2013). Note that the method used by Romberg et al. (2013) to achieve this performance involves a highly more complex system than ours. However, our method has better recall which is the most important measure here since precision is already almost perfect.

In Figure 13 we present some examples of classified images with our model recognising 1,467 brands and logos. The query image is positioned at the top of each figure and the concept returned with the highest confidence at the bottom.



http://i419.photobucket.com/albums/pp273/monochorme/monochrome/Audemars%20Piguet/Audemars-Piguet-Royal-Oak-Jumbo-2012-2.jpg

audemarspiguet (89%)

(a)

http://s3.e-monsite.com/2010/09/11/05/resize_550_550//Picture-12.jpg

camel (74%)

(b)

http://www.uaeinteract.com/news/article_pics/uae-65328.jpg

etihad_airways (67%)

(c)

Figure 13. Illustration of logo recognition results for the following objects: Audemarspiguet watch, Camel pack of cigarettes; Etihad plane.

4.4. Implementation and usage

The module has been implemented in C++ and is ready for integration. To this purpose, we provide a library compiled with GCC under the x86_64 linux system. The only dependence to an external library is that to Caffe (which has several dependencies itself) to be able to extract features from CNN.

To test which logos are found into the database, the following commands should be used:

```
extract_features.bin model parameters.txt layer nbatches query.txt

direct_search  query.txt  base.txt  D K
```

where parameters.txt contains the file with the path of all test images and the rest of the necessary parameters. Further documentation of the method usage is given below:

Program	**Description**
extract_features.bin	Compute the signature
direct_search	Search into the database
File	**Description**
parameters.txt	Parameter file to compute the signature
base.txt	The database
model	The CNN model
query.txt	Features of query images

4.5. Next steps

The results obtained with CNN-based logo detection are very promising since they clearly outperform previous state of the art results reported in (Romberg et al., 2013). Following these authors' observation about the fact that logos are usually small objects in the image, we will focus on the exploitation of automatic object localisation in images to improve the performance of our CNN-based method. Another important research axis concerns the noise inherent to logo representations obtained from the Web. We noticed that the level of noise varies a lot among the different logos and we will work on semi-automatic methods for noise reduction, concentrating on logos that have a low quality representation in the current dataset.

5.Image based location detection

5.1. Related work

Related work for image based location detection was described in some detail in D5.2. Here we remind the reader that, as it is the case for a large variety of visual content mining tasks, CNNs were successfully used for location-related datasets (Babenko et al., 2014).

5.2. Method description

The same training process described in D5.2 for POI recognition with CNN is implemented here. The main difference is that AlexNet (Krizhevsky et al., 2012), the CNN architecture from D5.2 is replaced with the VGG architecture introduced by (Simonyan and Zisserman, 2014). Training is done with a dataset of 990 POIs and approximately 1,200 images per POI. Following the evaluation results from D5.2, which show that image re-ranking has no positive influence in this setting, the raw dataset collected from the Web is used here. Fine-tuning from the ImageNet model proposed in (Simonyan and Zisserman, 2014) toward POIs is used in order to speed-up the training process. The outputs of the fc7 layer (4096 dimensions) were compressed to 128 using a PCA matrix learned from a subset of 250,000 images of the CNN training set and used to compute image similarities.

5.3. Evaluation and testing

The CNN model for image location is tested in an image-based location detection task, as this was the case for VLAD and GVR features in D5.2. The method is tested on the MediaEval 2015 Placing Task dataset, which contains 931,573 test images and 4.7 million geotagged images as ground truth. After computing the similarity between the query image and the collection images, the top k neighbours are retained in order to predict candidate locations. We apply a simple incremental spatial clustering scheme in which the jth image is attributed to an existing cluster if it is within a 1 km range from any of the j-1 images that were already seen. In the end, the largest cluster is selected and its centroid is used as location estimate. If two clusters have the same size, the one that contains the top ranked image among the top k is selected.

Measure acc@km	Ours	IMCUBE (Kelm et al., 2015)	RECOD (Li et al., 2015)
0.01	0.08	0	0.01
0.1	1.76	0	0.09
1	5.19	0.02	0.44
10	7.43	0.18	1.99
100	9.07	0.54	3.57

Table 4. Image based location prediction results with different methods proposed for MediaEval 2015 Placing Task evaluation campaign. The evaluation measure is the % of accurately geotagged images at every precision range given in kilometers.

Following Placing Task 2015 instructions, results are reported for different geotagging accuracies. The results are presented in Table 4, along with those of competing image-based location methods tried during the evaluation campaign are used as baseline:

- IMCUBE (Kelm et al., 2015) – visual similarity that relies on a wide spectrum of visual features that relate to color and texture of the images. A kd-tree is used to speed-up the retrieval process.
- RECOD (Li et al., 2015) – visual similarity was based on BIC features.

The results presented in Table 4 confirm those obtained in D3.2 in that CNN based characterisation of images outperforms other types of features. Here, they are compared with global features of roughly the same dimensionality and the results obtained with the proposed method clearly outperform the baselines.

5.4. Implementation and usage

The implementation is identical to that of the POI recognition module presented in D5.2. Naturally, the recognition model and the configuration file of the neural network are updated to include the changes described in subsection 5.2.

5.5. Next steps

This task is now considered nearly complete and the tool will only marginally evolve until the end of the project. Given that overall accuracy seems low, only images whose localisation is done with high confidence will be shown to the users. Preliminary results show that it is possible to automatically determine confidence and this topic will be further investigated. Naturally, assistance will be provided with integration on a per need basis.

6. Conclusions and future work

During the second iteration of the project, work on developing visual mining and linking modules has continued in a sustained manner. We have kept abreast with the latest developments in computer vision in order to ensure high performance of the developed tools. Particular focus was put on developing methods that are adapted to the overall USEMP vision, i.e. empowering OSN users and to concrete project requirements. Pursuing initial work from D5.2, we continued work on scaling-up the learning processes through: (1) the use of Web corpora instead of manually annotated datasets and (2) adaptation of recognition models to domains of interest for the project. Work on concept detection continued with the investigation of the influence of noise on performance and the proposal of semantic features that are adapted to individual image content. A new line of work, that exploits concept detection, concerned the private/non-private image classification. Preliminary results are very encouraging but also show that a lot of progress is still to be made, mainly concerning the personalization of classification models in this highly subjective task. For logo detection, we adapted deep learning methods and showed that they perform significantly better than the bag of visual words methods tested in D5.2. Finally, we have updated the image based location recognition module to include more powerful convolutional neural network architectures.

WP5 modules were informed by upstream work in WP2, WP3 and WP4 and are developed in close collaboration with WP6 work on disclosure scoring and setting. Equally important, assistance was provided with the integration of the modules in the DataBait architecture that is implemented as part of WP7. Concept detection was integrated for the pre-pilot studies and received positive feedback from users. Logo detection was integrated for the first pilot and will be evaluated in January and February 2016. The other modules are awaiting integration and should be ready for use before the final pilot of the project.

In the remaining months of USEMP, work will be focused on further improving some of the tools, including private/non-private image classification and logo recognition, and on the integration of all modules with text mining methods developed in T5.1. In particular, a new version of the face recognition tool will be provided and will be based on the use of CNNs. More generally, support will be provided until the end of the project in order to ensure a seamless integration of visual mining modules in DataBait.

7.References

Babenko, A., A. Slesarev, A. Chigorin, V. Lempitsky. (2014). Neural Codes for Image Retrieval. Proceedings of ECCV 2014.

Bergamo, A., & Torresani, L. (2012). Meta-class features for large-scale object categorization on a budget. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on (pp. 3085-3092). IEEE.

Branson, S., G. Van Horn, S. Belongie, P. Perona. (2014). Bird Species Categorization Using Pose Normalized Deep Convolutional Nets. Proceedings of British Machine Vision Conference (BMVC14)

Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). LIBLINEAR: A library for large linear classification. The Journal of Machine Learning Research, 9, 1871-1874.

Friedland, G., & Sommer, R. (2010). Cybercasing the Joint: On the Privacy Implications of Geo-Tagging. In HotSec.

Ginsca, A. L., Popescu, A., Le Borgne, H., Ballas, N., Vo, P., & Kanellos, I. (2015). Large-Scale Image Mining with Flickr Groups. In MultiMedia Modeling (pp. 318-334). Springer International Publishing.

R. Girshick, J. Donahue, T. Darrell, J. Malik. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR14)

Hastie, T., Friedman, J., & Tibshirani, R. (2001). The elements of statistical learning (Vol. 1). Springer, Berlin: Springer series in statistics.

Jia, Y. (2013). Caffe: An open source convolutional architecture for fast feature embedding.

Kelm, P., Sebastian Schmiedeke, Lutz Goldmann. Imcube @ MediaEval 2015 Placing Task: Hierarchical Approach for Geo-referencing Large-Scale Datasets. Working notes of MediaEval 2015.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

Li, L.-T., Javier A.V. Muñoz, Jurandy Almeida, Rodrigo T. Calumby, Otávio A. B. Penatti, Ícaro C. Dourado, Keiller Nogueira, Pedro R. Mendes Júnior, Luís A. M. Pereira, Daniel C. G. Pedronette, Jefersson A. dos Santos, Marcos A. Gonçalves, Ricardo da S. Torres. RECOD @ Placing Task of MediaEval 2015. Working notes of MediaEval 2015.

Minkus, T., Liu, K., & Ross, K. W. (2015). Children Seen But Not Heard: When Parents Compromise Children's Online Privacy. In Proceedings of the 24th International Conference on World Wide Web (pp. 776-786). International World Wide Web Conferences Steering Committee.

Paine, C., Reips, U. D., Stieger, S., Joinson, A., & Buchanan, T. (2007). Internet users' perceptions of 'privacy concerns' and 'privacy actions'. International Journal of Human-Computer Studies, 65(6), 526-536.

A.S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson. (2014). CNN Features off-the-shelf: an Astounding Baseline for Recognition. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR14), DeepVision Workshop

S. Romberg, L. Garcia Pueyo, R. Lienhart, R. van Zwol. (2011). Scalable Logo Recognition in Real-World Images. Proceedings of ACM International Conference on Multimedia Retrieval 2011 (ICMR11), Trento

S. Romberg, R. Lienhart. (2013). Bundle min-Hashing For Logo Recognition. Proceedings of ACM International Conference on Multimedia Retrieval 2013 (ICMR13), Dallas

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei. (2014). ImageNet Large Scale Visual Recognition Challenge. arXiv technical report: http://arxiv.org/abs/1409.0575

P. Sermanet, Y. LeCun. (2011). Traffic Sign Recognition with Multi-Scale Convolutional Networks. Proceedings of International Joint Conference on Neural Networks (IJCNN11)

Simonyan, K., and A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

Spyromitros-Xioufis, E., Papadopoulos, S., Kompatsiaris, I. Y., Tsoumakas, G., & Vlahavas, I. (2014). A comprehensive study over vlad and product quantization in large-scale image retrieval. Multimedia, IEEE Transactions on, 16(6), 1713-1728.

Squicciarini, A. C., Caragea, C., & Balakavi, R. (2014). Analyzing images' privacy for the modern web. In Proceedings of the 25th ACM conference on Hypertext and social media (pp. 136-147). ACM.

Y. Sun, X. Wang, X. Tang. (2014). Deep Learning Face Representation from Predicting 10,000 Classes. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR14)

L. Yang, P. Luo, C. C. Loy, X. Tang. (2015). A Large-Scale Car Dataset for Fine-Grained Categorization and Verification. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR15)

R. Wu, S. Ya, Y. Shan, Q. Dang, G. Sun. (2015). Deep image: Scaling up image recognition. arXiv technical report:

You, Q., Bhatia, S., & Luo, J. (2015). A Picture Tells a Thousand Words--About You! User Interest Profiling from User Generated Visual Content. arXiv preprint arXiv:1504.04558.

Zerr, S., Siersdorfer, S., Hare, J., & Demidova, E. (2012). I Know What You Did Last Summer!: Privacy-Aware Image Classification and Search. SIGIR.

B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, A. Oliva. (2014). Learning Deep Features for Scene Recognition using Places Database. In Advances in Neural Information Processing Systems (NIPS 2014)