Photo Privacy Detection based on Text Classification and Face Clustering

Lyudmila Kopeykina
National Research University Higher School of Economics
Nizhny Novgorod, Russia
lnkopeykina@mail.ru

Andrey Savchenko
Laboratory of Algorithms and Technologies for Network Analysis,
National Research University Higher School of Economics
Nizhny Novgorod, Russia
avsavchenko@hse.ru

***Abstract*— Nowadays, the photo privacy detection is becoming an acute task due to a wide spread of mobile devices with photos published on social networks. As a photo might contain private or sensitive data, there is an urgent need to accurately determine them and impose restrictions on their processing. In this paper we focus on the task of personal data detection in a photo gallery. A novel two-stage approach is proposed. At first, text of scanned documents is processed based on an EAST text detector, and extracted text is recognized using Tesseract and neural network classifier. At the second stage, face clustering is implemented for the remaining photos to identify large groups of people (friends, relatives) whose photos also refer to personal data and must be processed directly on a mobile device. The remaining images can be sent to a remote server for processing with higher accuracy. The experimental results of text recognition and face clustering methods using various convolutional networks for facial features extraction are presented.**

Keywords—photo privacy detection, face clustering, text detection and classification

I. Introduction

The photo gallery of a typical mobile device contains unique information about its user and reflects his or her preferences [1]. As a result, image-processing methods can be applied to build visual recommender engines [2]. Such deep learning-based methods usually require significant computing resources and should be implemented on a remote server with GPUs. However, there is an urgent need to restrict the processing of photos with some sensitive data in order to avoid the potential risk of inappropriate usage of private information.

The privacy detection on photos is a worth considering problem [3, 4] that has already reached a certain level of maturity [5, 6, 7]. The demand for handling this issue is justified by the need to distinguish personal photos that cannot be transferred to the third parties in terms of privacy policy, and public information that can be sent to a remote server for further deep processing and analysis. Moreover, the separate processing of public and private photos improves the accuracy and computational efficiency of algorithms.

It is noticeable that the vast majority of private images mainly contain such characteristics like human faces, textual data (identification data and credit card numbers) and other general objects (private cars and buildings) [3, 8]. Therefore, this work proposes a unified approach for personal data detection in photo gallery using well-known methods of face classification [9, 10, 11] and text recognition (optical character recognition, OCR) [12, 13]. In particular, to detect scanned personal documents, it is proposed to sequentially use the EAST text detector [14], the Tesseract OCR library [12] and the neural network classification of recognized text on images. To detect personal photos containing faces of the user himself, his close friends and relatives, the well-known methods of face clustering [15, 16, 17] are applied to face embeddings extracted with CNNs (convolutional neural networks) [2, 18].

The rest of the paper is organized as follows: In Section II we describe the proposed approach in detail. Section III includes experimental study of privacy detection methods. Finally, in Section 4 the conclusion and future plans are discussed

II. Materials and Methods

In this paper we concentrate on the following task. It is required to assign an image from photo album to one of two possible classes: private or public. The proposed approach is shown in Fig. 1. Let us discuss the most important parts of this pipeline in the rest of this section.

A. Detection of Scanned Documents

As a part of scanned documents detection, it is proposed to consider various methods of text recognition. Firstly, image areas containing textual information are detected using the EAST algorithm [14]. Further, Tesseract OCR in image_to_string mode with LSTM (Long-Short Term Memory) recursive model is used to recognize text in each detected area. The given approach is subsequently compared with a simplified text recognition method, in which the step of preliminary text detection by the EAST detector is omitted. Instead, Tesseract is used both in text recognition mode and in automatic page segmentation mode.

After that, to classify personal data in the extracted text, it is proposed to use a neural network, which is trained based on the input sequence of words recognized in the training set of scanned documents [13]. One-hot encoding is used to represent the input data as a feature vector. To be more exact, a dictionary of the V most frequently used words in the training set is created, and each text is represented as a V-dimensional binary vector, where the v-th component of the vector is 1 only if the v-th word from the dictionary is presented in the input text (so-called bag-of-words model) [19, 20].

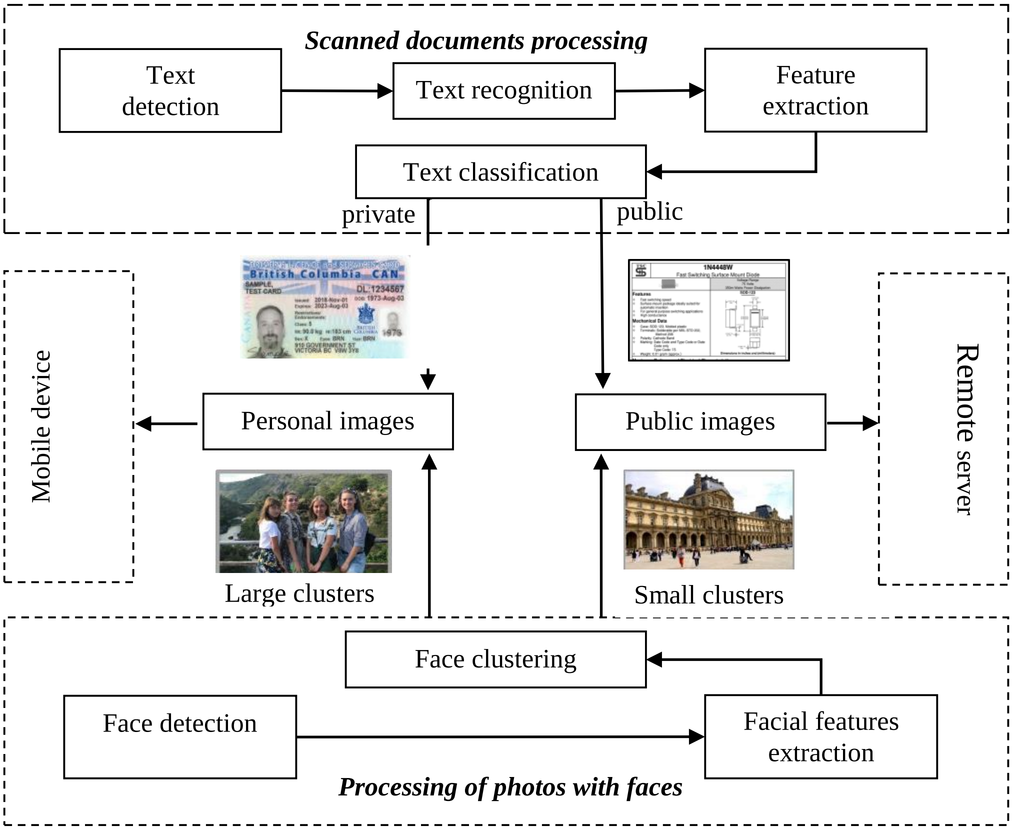

Fig. 1. Proposed pipeline for photo privacy detection.

To solve the binary classification problem, it is proposed to use a computationally efficient implementation of a fully connected neural network, which has already shown high performance in a similar problem of sentiment analysis [19]. To train the above-mentioned network, we created a balanced corpus of 700 images [13]. The positive class is presented by 350 images of driving license and medical insurance cards, passports and invoices from extension of the MIDV dataset [21], whereas negative class consists of photos from publicly available datasets for text classification tasks DIQA [22] and Ghega [23]. This approach is sometimes as accurate as more complex methods based on CNNs and LSTMs. Moreover, it outperforms well-known traditional methods for detecting personal data, for example, the keyword spotting method [13].

B. *Detection of Personal Photos Based on Face Clustering*

As scanned documents are not the only option for personal data in the gallery, it is proposed to select images that contain faces of the user himself, his close friends and relatives [1, 24]. To detect such kind of personal photos, it is proposed to apply the following approach. At first, the facial regions are detected in all photographs using well-known methods for face detection like cascade classifiers or MTCNN [25]. Since there are no labels of people in the user's photo gallery, the task can be reformulated as a face clustering problem [16, 24]. For doing this, D-dimensional feature vectors are extracted [9, 11] for each of $N > 0$ selected facial images by using a CNN, pre-trained to identify faces from a large (external) datasets like VGGFace-2, MS-Celeb, etc.

The procedure for combining selected individuals into clusters supposes the assignment of each i-th facial image ($i = 1, ..., N$) to one of C $\geq$ 1 group, where C is usually unknown. Hence, one can apply either traditional agglomerative clustering algorithms or rank linkage [15, 16] and graph CNNs [17]. An image is considered to be private if it contains faces from sufficiently large clusters. In other words, a person presents at least $K_{\min}$ times on different types of photos, where $K_{\min}$ is a hyper-parameter of our method. That assumption is based on the idea that the user's gallery contains his own face and faces of his close friends on the substantial part of photos.

III. Experiments and Results

In this section we present the experimental results of a comparative analysis of the well-known text classification. Moreover, the comparison of clustering methods applied to facial features extracted with various CNN is given. Finally, we analyze the performance of our approach to split user's photos into to private and public images.

A. *Detection of Scanned Documents*

At first, we compare various approaches for text extraction in terms of traditional keyword spotting method, which aims to search specially selected words ("passport", "card", etc.) [13] in recognized text. Namely, we compare simultaneous detection of text on images and its recognition using Tesseract with the approach when text regions are preliminary detected by EAST detector and text is recognized by Tesseract OCR engine. In addition to traditional keyword spotting, three neural network models are compared:

- Recurrent model, which fed a sequence of 400 words from a dictionary of $V = 5000$ frequently encountered words as input for the vector representation (embedding) with the size of the attribute space 256. Next, we use the LSTM layer with 128 hidden components, the dropout layer with a drop rate of 0.5.

- CNN, consisting of one-dimensional convolutional layer (with 32 neurons, core size of 7 and ReLU activation function), maxpooling and dropout layers (with a drop rate of 0.5). As the first layer of the model, a vector representation (embedding) of 256 was also used.

- Fully connected network with 2 hidden layers of 16 neurons with hyperbolic tangent activation. The V-dimensional vector encoded as described in Subsection IIA (bag-of-words) is considered as input for the model.

The last fully connected layer of each model used the sigmoid activation. To train classifiers, TensorFlow and Keras frameworks were used. All classifiers were trained over 20 epochs using the RMSprop optimizer.

A quantitative comparison of all methods described above is presented in Table I. The results were obtained using a 5-fold cross-validation.

TABLE I. RESULS FOR CLASSIFICATION OF SCANNED DOCUMENTS

	Model	**Precision**	**Recall**	**F-score**	**Error rate**
Tesseract	Keyword spotting	0.83	0.62	0.70	0.276
	LSTM	0.97	0.93	0.94	0.043
	CNN	0.88	0.77	0.82	0.161
	Fully-connected	0.98	0.94	0.95	0.028
Proposed (EAST+ Tesseract)	Keyword spotting	0.90	0.75	0.81	0.161
	LSTM	0.93	0.99	0.95	0.038
	CNN	0.89	0.79	0.83	0.144
	Fully-connected	1.00	0.97	0.98	0.015

Here the use of EAST text detector to identify areas with text was a reasonable solution. While the error rate attained using only Tesseract is more than 27%, the proposed preliminary detection of text using the EAST detector reduces this error to approximately 16%. In addition, we can conclude that the proposed implementation with the EAST text detector increases the average accuracy by approximately 2%. A fully-connected network achieves best results with accuracy that exceeds even traditional LSTM. Moreover, such an implementation 15% more accurately determines the image class of the document in comparison with the traditional keyword spotting.

B. *Face Clustering*

We used the publicly available facial datasets:

- Gallagher collection person dataset [26], which contains 589 images with 931 labeled faces of 32 various people. As only eyes positions are available in this dataset, to gather faces MTCNN [25] was preliminarily used to detect faces and choose the subject with the largest intersection of facial region with given eyes region. If the face is not detected, a square region with the size chosen as a 1.5-times distance between eyes is extracted.

- Subset of labeled faces in the wild (LFW) dataset [27] used to test face identification algorithms [11]. It includes photos of those subjects, who has at least two images in the original LFW dataset and at least one video in the YouTube Faces (YTF) collection.

Firstly, hierarchical agglomerative clustering is considered for the distance L_2 between normalized feature vectors with the following types of linkage: single linkage, average linkage, complete linkage, weighted linkage, centroid linkage and median linkage from the SciPy library. Further, the rank-order clustering [15] was examined as it was specially developed for organizing faces in photo albums. It uses special rank linkage, which is further used to compute distance measure. Then this approach was compared to the approximate rank-order algorithm [28], in which only the top-k neighbors are taken into consideration rather than the complete list of neighbors. This approach makes the actual rank of neighbors irrelevant because the importance is shifted towards the presence / absence of shared nearest neighbors. Finally, we examined clustering method based on the graph CNN [29, 30]. Each element of the feature matrix is considered as a separate vertex of the graph. Using the cosine distance, k nearest neighbors are found for each element of the dataset. Thus, by connecting between neighbors, a similarity graph for the entire dataset is obtained. Instead of processing such graph directly, subgraphs-proposals are first generated, on the basis of which the resulting clusters are subsequently built.

To extract facial features, traditional pre-trained models downloaded from the official websites of their developers were considered:

- VGGFace (VGGNet-16) [31] extracts 4096-D vectors;

- VGGFace2 (ResNet-50) [9] extracts 2048-D vectors;

- MobileNet [24] extracts 1024-D vectors;

- InsightFace (ArcFace) [32] extracts 512-D vectors;

- FaceNet (Inception ResNet v1) [10] extracts 512-D vectors.

Table III contains the Rand index (ARI), mutual information index (AMI), homogeneity and completeness. In addition, the average number K of selected clusters to the number of groups C and the b-cubed F-measure, traditional for assessing the quality of face clustering, are calculated.

Considering the results, clustering applied to facial features extracted with ResNet-50 (VGGFace2) and Inception ResNet v1 (FaceNet) perform more accurate results according to most of the metrics compared to other models. Although MobileNet is slightly inferior, it takes twice less time to extract face embeddings compared to VGGFace2 and FaceNet. InsightFace features in most cases shows slightly worse capacity to define clusters. In addition, the weighted linkage demonstrates higher F-score for both datasets in comparison with other clustering methods (over 92%).

TABLE II. CLUSTERING RESULTS FOR GALLAGHER DATASET

	CNN	Time, sec	K/C	ARI	AMI	Homogeneity	Completeness	F-score
Rank-order	VGGFace2	32.17	1.25	0.480	0.627	0.794	0.635	0.706
	VGGFace	21.72	1.50	0.439	0.569	0.764	0.585	0.671
	MobileNet	22.71	2.09	0.674	0.678	0.965	0.611	0.725
	InsightFace	27.84	1.59	0.502	0.530	0.729	0.716	0.625
	FaceNet	24.54	1.53	0.674	0.681	0.906	0.633	0.760
Single linkage	VGGFace2	0.016	3.06	0.267	0.568	0.553	0.752	0.631
	VGGFace	0.024	2.75	0.260	0.559	0.531	0.763	0.623
	MobileNet	0.022	2.72	0.280	0.586	0.562	0.767	0.636
	InsightFace	0.025	2.72	0.109	0.294	0.296	0.607	0.503
	FaceNet	0.013	3.09	0.286	0.592	0.579	0.762	0.642
Average linkage	VGGFace2	0.021	1.50	0.662	0.763	0.762	0.819	0.892
	VGGFace	0.021	2.15	0.648	0.771	0.794	0.808	0.802
	MobileNet	0.019	2.03	0.882	0.868	0.961	0.822	0.891
	InsightFace	0.027	3.12	0.707	0.711	0.891	0.660	0.739
	FaceNet	0.018	2.31	0.886	0.868	0.942	0.835	0.895
Complete linkage	VGGFace2	0.032	1.09	0.859	0.867	0.911	0.853	0.888
	VGGFace	0.023	1.18	0.616	0.743	0.876	0.690	0.711
	MobileNet	0.019	0.41	0.863	0.816	0.798	0.861	0.836
	InsightFace	0.018	1.75	0.367	0.576	0.819	0.521	0.512
	FaceNet	0.013	0.65	0.710	0.813	0.826	0.830	0.821
Weighted linkage	VGGFace2	0.033	1.50	0.891	0.898	0.946	0.876	**0.921**
	VGGFace	0.019	1.03	0.599	0.737	0.704	0.830	0.762
	MobileNet	0.018	0.75	0.751	0.788	0.792	0.818	0.806
	InsightFace	0.018	1.72	0.655	0.697	0.806	0.675	0.734
	FaceNet	0.015	1.47	0.884	0.881	0.934	0.857	**0.902**
Approximate rank-order	VGGFace2	0.785	3.91	0.515	0.535	0.586	0.641	0.704
	VGGFace	1.312	3.78	0.446	0.485	0.509	0.681	0.653
	MobileNet	1.414	6.68	0.417	0.516	0.522	0.795	0.635
	InsightFace	1.220	5.78	0.324	0.324	0.471	0.656	0.571
	FaceNet	1.092	4.05	0.567	0.621	0.626	0.764	0.724
GCN-D	VGGFace2	5.006	1.67	0.867	0.845	0.954	0.793	0.859
	VGGFace	4.741	0.78	0.641	0.536	0.627	0.539	0.578
	MobileNet	6.290	0.69	0.675	0.748	0.799	0.742	0.728
	InsightFace	6.862	0.65	0.409	0.612	0.603	0.682	0.637
	FaceNet	6.164	0.91	0.636	0.726	0.751	0.749	0.687

TABLE III. CLUSTERING RESULTS FOR LFW DATASET

	CNN	Time, sec	K/C	ARI	AMI	Homogeneity	Completeness	F-score
Rank-order	VGGFace2	416.73	0.96	0.719	0.781	0.980	0.911	0.862
	VGGFace	309.44	0.82	0.675	0.748	0.812	0.762	0.746
	MobileNet	305.03	0.77	0.786	0.816	0.944	0.907	0.806
	InsightFace	361.02	1.21	0.673	0.721	0.842	0.912	0.683
	FaceNet	359.62	0.91	0.784	0.832	0.924	0.917	0.812
Single linkage	VGGFace2	0.47	1.66	0.969	0.940	0.998	0.951	0.917
	VGGFace	0.64	1.86	0.854	0.876	0.962	0.931	0.847
	MobileNet	0.60	1.52	0.744	0.871	0.930	0.951	0.854
	InsightFace	0.68	2.08	0.837	0.838	0.951	0.911	0.804
	FaceNet	0.50	1.63	0.967	0.935	0.993	0.952	0.912
Average linkage	VGGFace2	0.69	1.49	0.966	0.945	0.998	0.955	**0.926**
	VGGFace	0.61	1.36	0.946	0.933	0.988	0.953	0.911
	MobileNet	0.64	1.48	0.968	0.943	0.997	0.954	0.923
	InsightFace	0.73	1.37	0.887	0.873	0.972	0.920	0.831
	FaceNet	0.67	1.54	0.960	0.937	0.997	0.949	**0.918**
Complete linkage	VGGFace2	0.57	1.13	0.744	0.935	0.992	0.951	0.910
	VGGFace	0.62	0.99	0.621	0.873	0.966	0.921	0.821
	MobileNet	0.62	1.06	0.852	0.925	0.980	0.953	0.894
	InsightFace	0.55	0.90	0.756	0.793	0.926	0.889	0.720
	FaceNet	0.53	1.07	0.748	0.929	0.986	0.951	0.900

TABLE III. CLUSTERING RESULTS FOR LFW DATASET (CONT.)

Weighted linkage	VGGFace2	0.63	1.37	0.893	0.941	0.998	0.952	**0.923**
	VGGFace	0.61	1.28	0.925	0.925	0.984	0.950	0.901
	MobileNet	0.59	1.44	0.961	0.940	0.996	0.952	0.919
	InsightFace	0.67	1.42	0.879	0.864	0.972	0.913	0.820
	FaceNet	0.64	1.44	0.935	0.938	0.997	0.950	**0.919**
Approximate rank-order	VGGFace2	9.49	1.42	0.803	0.877	0.924	0.952	0.923
	VGGFace	7.12	1.30	0.621	0.706	0.893	0.816	0.724
	MobileNet	7.06	1.79	0.610	0.741	0.864	0.912	0.740
	InsightFace	12.32	1.57	0.684	0.711	0.849	0.908	0.685
	FaceNet	12.72	1.13	0.782	0.859	0.932	0.937	0.844
GCN-D	VGGFace2	30.33	0.84	0.075	0.395	0.814	0.711	0.512
	VGGFace	28.47	0.69	0.044	0.235	0.866	0.669	0.456
	MobileNet	31.23	0.86	0.332	0.665	0.882	0.825	0.639
	InsightFace	30.18	0.74	0.802	0.732	0.874	0.875	0.666
	FaceNet	31.79	0.92	0.141	0.543	0.828	0.770	0.588

Agglomerative clustering with average linkage performs the second most accurate results (approximately 90%). Furthermore, connectivity graph-based method demonstrates poor results on the given data. The use of rank distance is impractical due to the rather low values for each metric and its quadratic complexity. Even though the approximation of rank-order clustering takes less time to split data into groups compared to the original method, the results still do not outperform those of traditional agglomerative algorithms.

Moreover, we analyzed the dependence between the minimum number of faces in cluster to set it private (K_{min}) and the type 1 and type 2 error rates for the LFW subset (Fig. 2). Since ground truth labels in terms of private and public photos for that dataset were not provided, we determined them as follows. All objects from classes, the number of photos in which is greater than or equal to K_{min}, were considered to be private. The remaining images were assigned to public images. We used agglomerative clustering with weighted linkage and VGGFace2 descriptor as it provided best results according to conducted experiments.

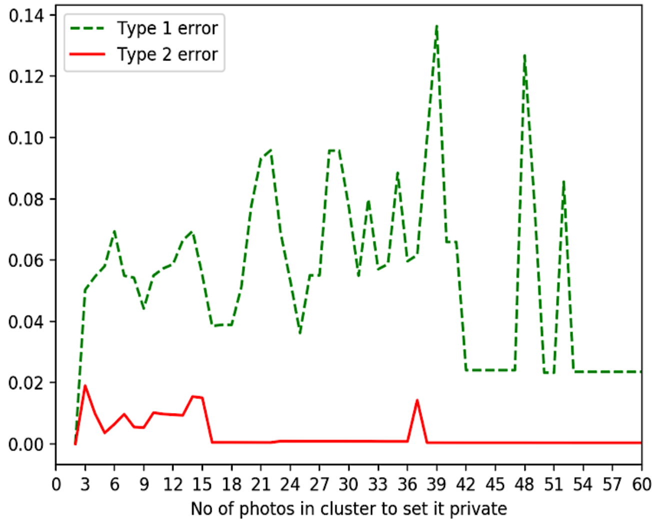

Fig. 2. The dependence between the minimal number K_{min} of photos in a personal cluster and type1/type 2 error rates, LFW dataset.

According to the results, zero rate of missing private photos is achieved with K_{min}=2. It means that all photos from dataset are initially private and they are marked as private by algorithm. If K_{min}=3, then 5% of private photos will be moved to public set. With an increase of K_{min}, the trend for type 1 error is going upwards unstably and ends up with 2%. At the same time, the probability to assign public images to private decreases and reaches 0%.

In the final experiment, we compared the results given by various descriptors on LFW (Table IV). “0” class consists of 3263 private images, whereas public class “1” includes 474. Here, images containing faces from clusters that include K_{min}=3 or more facial images, were considered personal. Here all face descriptors lead to a fairly high quality of detection, but zero probability of missing personal data was not achieved. In this case, the best results are obtained using VGGFace2 (ResNet-50) and FaceNet models.

TABLE IV. CLASSIFICATION RESULTS FOR LFW

Feature extractor	FPR	FNR	Precision	Recall	F1-score	Error rate
VGGFace2	0.051	0.019	0.738	0.978	0.842	0.047
VGGFace	0.055	0.276	0.655	0.723	0.688	0.084
MobileNet	0.054	0.168	0.687	0.831	0.752	0.069
InsightFace	0.115	0.281	0.474	0.719	0.571	0.137
FaceNet	0.056	0.044	0.712	0.952	0.816	0.055

IV. CONCLUSION

The task of personal photos detection is difficult in terms of finding an effective solution due to its inherent subjectivity. In this paper, it is assumed that personal data contains confidential textual information and images with the user, his close friends and relatives. This assumption allows to highlight personal photos accurately and impose restrictions on their processing. To highlight such data, a novel approach was proposed in the current work (Fig. 1). It is proposed to use the EAST text detector and recognize text in the detected areas with Tesseract OCR library to classify scanned documents. It has been experimentally shown that a simple fully-connected neural network for text encoded using bag-of-words [13] exceeds more complex network architectures, such as CNN, by more than 10% and achieves high accuracy in detecting personal documents. In addition, in agglomerative clustering with a weighted linkage performed higher results in extracting groups of user’s faces, friends and relatives (Tables II and III).

Acknowledgment

The paper was prepared within the framework of the Academic Fund Program at the National Research University Higher School of Economics (HSE University) in 2019-2020 (grant No 19-04-004) and by the Russian Academic Excellence Project «5-100».

References

[1] I. Grechikhin and A.V. Savchenko, “User Modeling on Mobile Device Based on Facial Clustering and Object Detection in Photos and Videos,” Iberian Conference on Pattern Recognition and Image Analysis, Springer, Cham, pp. 429-440, 2019.

[2] I. Goodfellow, Y. Bengio and A. Courville, “Deep learning,” MIT press, 2016.

[3] L. Tran, D. Kong, H. Jin and J. Liu, “Privacy-CNH: A framework to detect photo privacy with convolutional neural network using hierarchical features,” Thirtieth AAAI Conference on Artificial Intelligence (AAAI), pp. 1317-1323, 2016.

[4] H. Zhong, A.C. Squicciarini, D.J. Miller and C. Caragea, “A Group-Based Personalized Model for Image Privacy Classification and Labeling,” International Joint Conferences on Artificial Intelligence (IJCAI), vol. 17, pp. 3952-3958, 2017.

[5] A. Tonge and C. Caragea, “Dynamic deep multi-modal fusion for image privacy prediction,” The World Wide Web Conference (WWW), pp. 1829-1840, 2019.

[6] A. Tonge and C. Caragea, “Image privacy prediction using deep neural networks,” ACM Transactions on the Web (TWEB), vol. 14, no. 2, pp. 1-32, 2020.

[7] C. Sitaula, Y. Xiang, S. Aryal and X. Lu, “Unsupervised deep features for privacy image classification,” Pacific-Rim Symposium on Image and Video Technology, pp. 404-415, 2019.

[8] J. He, B. Liu, D. Kong, X. Bao, N. Wang, H. Jin and G. Kesidis, “Puppies: Transformation-supported personalized privacy preserving partial image sharing,” 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), IEEE, pp. 359-370, 2016.

[9] Q. Cao, L. Shen, W. Xie, O.M. Parkhi and A. Zisserman, “Vggface2: A dataset for recognising faces across pose and age,” 3th International Conference on Automatic Face & Gesture Recognition (FG), IEEE, pp. 67-74, 2018.

[10] F. Schroff, D. Kalenichenko and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 815-823, 2015.

[11] A.V. Savchenko and N.S. Belova, "Unconstrained face identification using maximum likelihood of distances between deep off-the-shelf features," Expert Systems with Applications, vol. 108, pp. 170-182, 2018.

[12] R. Smith, “An overview of the Tesseract OCR engine” Ninth International Conference on Document Analysis and Recognition (ICDAR), IEEE, vol. 2, pp. 629-633, 2007.

[13] L. Kopeykina, A.V. Savchenko, “Automatic privacy detection in scanned document images based on deep neural networks,” Proceedings of International Russian Automation Conference (RusAutoCon), IEEE, pp. 1-6, 2019.

[14] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He and J. Liang, "EAST: an efficient and accurate scene text detector," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5551-5560, 2017.

[15] C. Zhu, F. Wen and J. Sun, “A rank-order distance based clustering algorithm for face tagging,” CVPR IEEE, pp. 481-488, 2011.

[16] Y. Shi, C. Otto and A. K. Jain, “Face clustering: representation and pairwise constraints,” IEEE Transactions on Information Forensics and Security, vol. 13, no. 7, pp. 1626-1640, 2018.

[17] L. Yang, X. Zhan, D. Chen, J. Yan, C. C. Loy and D. Lin, “Learning to cluster faces on an affinity graph,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2298-2306, 2019.

[18] A.V. Savchenko, "Probabilistic neural network with complex exponential activation functions in image recognition," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 2, pp. 651-660, 2020.

[19] F. Chollet, "Deep learning with Python," Manning Publications, 2017.

[20] A.V. Savchenko and E.V. Miasnikov, “Event recognition based on classification of generated image captions,” International Symposium on Intelligent Data Analysis (IDA), pp. 418-430, 2020.

[21] V.V. Arlazarov, K. Bulatov, T. Chernov and V.L. Arlazarov, “MIDV-500: a dataset for identity document analysis and recognition on mobile devices in video stream”, Computer Optics, vol. 43, no. 5, pp. 818-824, 2019. DOI: 10.18287/2412-6179-2019-43-5-818-824.

[22] P. Ye and D. Doermann, “Document image quality assessment: A brief survey”, 12th International Conference on Document Analysis and Recognition, IEEE, pp. 723-727, 2013.

[23] A. Bartoli, G. Davanzo, E. Medvet and E. Sorio, “Improving features extraction for supervised invoice classification,” Proceedings of the 10th IASTED International Conference, vol. 674, no. 040, p. 401, 2010.

[24] A.V. Savchenko, “Efficient facial representations for age, gender and identity recognition in organizing photo albums using multi-output ConvNet,” PeerJ Computer Science, e197, 2019.

[25] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, 2016.

[26] A.C. Gallagher and T. Chen, “Clothing cosegmentation for recognizing people” IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8, 2008.

[27] G.B. Huang, M. Mattar, T. Berg and E. Learned-Miller, “Labeled faces in the wild: A database forstudying face recognition in unconstrained environments,” 2018.

[28] C. Otto, D. Wang and A.K. Jain, “Clustering millions of faces by identity,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 2, pp. 289-303, 2017.

[29] L. Yang, D. Chen, X. Zhan, R. Zhao, C.C. Loy and D. Lin, “Learning to cluster faces via confidence and connectivity estimation,” arXiv preprint arXiv:2004.00445, 2020.

[30] L. Yang, D. Chen, X. Zhan, R. Zhao, C.C. Loy and D. Lin, “Learning to cluster faces on an affinity graph,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2298-2306, 2019.

[31] O.M. Parkhi, A. Vedaldi and A. Zisserman, “Deep face recognition,” Britich Machine Vision Conference (BMVC), 2015.

[32] J. Deng, J. Guo, N. Xue and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4690-4699, 2019.