

# MIDV-DM: A Document-Oriented Dataset for Image Manipulation Detection and Localization

A.V. Chuiko<sup>1,2</sup>, I.A. Kunina<sup>1,2</sup>, S.A. Usilin<sup>1,2</sup>, C. Chen<sup>3</sup>, S. Tan<sup>3</sup>, D.P. Nikolaev<sup>1,2</sup>, V.V. Arlazarov<sup>1,2</sup>

<sup>1</sup> Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences,  
Prospekt 60-letii Oktiabria 9, Moscow, 119333, Russia;

<sup>2</sup> Smart Engines Service LLC, Prospekt 60-letii Oktiabria 9, Moscow, 117312, Russia;

<sup>3</sup> Shenzhen MSU-BIT University (SMBU), Shenzhen, Guangdong, China

## Abstract

As the scope of application of document recognition systems in business processes increases, so does the number of attacks on these systems. One form of such attacks could involve software for manipulating a digital image of a document. The development of methods for image manipulation detection and localization is complicated with the fact that available datasets neither contain images of documents nor have diversity in capture conditions and document types. Furthermore, these datasets do not cover the range of possible kinds of manipulations that occur under natural conditions. In this paper, we introduce MIDV-DM – a publicly available benchmark designed for the development and testing of methods aimed at detecting and localizing manipulations in identity document images. It contains images subjected to eight types of manipulations, which we have conceptually categorized based on our analysis of over 2000 real-world fraud attempts. In total, MIDV-DM contains 1000 original document images from the public MIDV-2020 dataset and 8000 automatically created manipulated images based on them, along with the ground truth masks and annotations. The paper also describes the process of obtaining baseline quality based on the IML-ViT model. The authors believe that MIDV-DM will open new opportunities for researchers to advance technologies for document image authenticity analysis.

**Keywords:** image manipulation detection, document forgery, copy-move, splicing, visible watermark, image forensic, document images, benchmark dataset.

**Citation:** Chuiko AV, Kunina IA, Usilin SA, Chen C, Tan S, Nikolaev DP, Arlazarov VV. MIDV-DM: A Document-Oriented Dataset for Image Manipulation Detection and Localization. *Computer Optics* 2025; 49(6): 1093-1101. DOI: 10.18287/COJ1768.

## Introduction

Automatic document recognition has remained relevant for many years. For example, recognition of identity documents may be required when receiving government and commercial services. However, as the scope of application of document recognition systems expands, the level of digital fraud also increases. The development and implementation of technologies for monitoring the authenticity of documents presented by the user will reduce losses from fraudulent activities.

Modern researchers have proposed a lot of approaches to verifying the authenticity of a document. Among different approaches are checking for the presence and the right location of the government seal stamps [1] and holograms [2, 3], the conformance of the fonts to the government standards [4], etc.

However, such approaches based on the analysis of document content cannot reliably detect traces resulting from digital editing of an image of a real document. Image manipulation detection and localization (IMDL) [5] methods aimed to identify edited regions from a suspicious image are rapidly developed [6,7]. However, such methods are usually evaluated on datasets such as CoMoFoD [8], MICC-F220 [9], CASIAv2 [10] that contain natural images. This approach makes the performance of such methods unpredictable when applied to document images. For example, copy-move forgery detection in an image of a document could most likely struggle in the multiple presence of the same characters with the same font at the same scale. Such objects called Similar but Genuine Objects (SGO) in the IMDL field and their pairs could have the same features as genuine and manipulated ones.

To evaluate copy-move forgery detection algorithms in the presence of SGO the authors of [11] proposed the COpy-moVe forgERY dAtabase with similar but Genuine objEcts (COVERAGE). However, this dataset contains natural images again, the forged regions are relatively large (average 23 % of pixels tampered) and the number of the images is relatively small (100 original–forged image pairs).

The problem of small forgeries (average of only 0.21 %) detection in the presence of many SGO in document images is studied in [12]. The authors proposed the Copy-Move ID (CMID) dataset that contains copy-move letters within the document image. Genuine images were acquired using a smartphone and cropped to mostly contain the document.

In [13] authors presented a corpus of corrupted grayscale payslips with forged characters. Three types of forgery were considered: copy-move, splicing and synthesis.

Being close to document text editing, in [14] the authors proposed FMIDV (forged mobile ID video dataset) based on photos, scans and templates from MIDV-2020 [15]. This dataset contained forged IDs with respect to guilloche patterns.

The authors considered such a scenario as a way to mask original data when fraudulent data being shorter than the original one.

In our work, we have analyzed over 2000 real-world fraud attacks of document recognition system based on their experts' descriptions. The expert descriptions were provided by partners specializing in document analysis and anti-fraud, who regularly investigate incidents in production-grade document recognition systems. Based on the analysis results, we came to the conclusion that manipulations with document text data do not exhaust all possible threats. For example, document image editing could imply manipulations with a holder's photo (see Fig. 3) or with the document itself (see Fig. 4 or Fig. 5). In addition, one of the most common attacks on the recognition system, due to its simplicity, is the use of images taken from the Internet. Usually, such images do not contain additional manipulation with their content, but could contain watermarks (see Fig. 8 or Fig. 9), different graphic elements (see Fig. 10) or holder's data masking (see Fig. 7). In total, we identified eight types of manipulations with a real document image, which is more than the number of types typically considered in existing datasets that have been created to develop IMDL methods on document images.

Therefore, in this paper we present MIDV-DM – a public dataset of genuine images and automatically created manipulated images based on them and subjected to eight types of manipulations, along with the ground truth masks and annotations. To estimate how challenging images MIDV-DM contains, we evaluate state-of-the-art IML-ViT model [16] on them.

The dataset is available for download at <ftp://smartengines.com/midv-dm/>.

### 1. Types of manipulations

Let us call the base image a genuine image that is manipulated, and the donor image an image used to modify the base one. As a result of the analysis, we identified the following types of manipulations that occur in natural conditions:

1. *Copy-move manipulation of characters (copy\_move)*. A region of a character from the base image is copied and inserted in another region of the base image. Thus, the base image contains the copied and inserted regions (see Fig. 1).
2. *Splicing of text fields (splicing\_symbol)*. A region of text field from the donor image is inserted in the corresponding region of the base image (see Fig. 2).
3. *Splicing of photo region (splicing\_photo)*. A region of a photo or a face from the donor image is inserted in the corresponding region of the base image (see Fig. 3).
4. *Splicing of the entire document (splicing\_document)*. A region of the entire document from the donor image is inserted in the corresponding region of the base image (see Fig. 4).
5. *Gluing (gluing)*. The base image and the donor one are merged along a horizontal or vertical line (see Fig. 5).
6. *Overlaying printed text (text\_overlay)*. Some text is overlaid on the base image using one of the digital fonts (see Fig. 6). Usually, before such a manipulation, there is masking of information in the region where the text is going to be overlaid.
7. *Masking of information (information\_masking)*. The region of the base image is masked using irreversible transformations (see Fig. 7).
8. *Overlaying foreign objects (foreign\_object)*. Objects not related to the document or the scene are overlaid on the base image. The objects could vary in their degree of transparency (see Fig. 8), in the presence of periodicity (see Fig. 9), by class, for example, an object can be text, an emblem, or vector graphics elements (see Fig. 10), etc.

In parentheses, we identified the types of manipulations which were used by us to separate the images with manipulation types during the creation of MIDV-DM.

In the next section we will describe the process of creating images that have been manipulated in the ways listed.

### 2. Creation of the MIDV-DM Dataset

To create MIDV-DM, we used the *photo* section of the public MIDV-2020 dataset [15] as the base images. This section contains 100 photographs of various documents of the following 10 types:

1. ID Card of Albania (alb\_id);
2. Passport of Azerbaijan (aze\_passport);
3. ID Card of Spain (esp\_id);
4. ID Card of Estonia (est\_id);
5. ID Card of Finland (fin\_id);
6. Passport of Greece (grc\_passport);
7. Passport of Latvia (lva\_passport);
8. Internal passport of Russia (rus\_internalpassport);
9. Passport of Serbia (srb\_passport);
10. ID Card of Slovakia (svk\_id);

Each document contains generated realistic text fields and faces. The photographs of the documents themselves were taken under varying lighting conditions, background textures, and projective distortions. Note that all these images were not subjected to any software manipulation. In parentheses, the document type identifiers are indicated, which correspond to the original identifiers from MIDV-2020 [15] and were used by us to separate the images by document types during the creation of MIDV-DM.

To perform manipulations of all types, except for two scenarios of overlaying foreign objects (more details in the corresponding section below), we used annotations from MIDV-2020 to first geometrically normalize the document image in such a way that the coordinates of its fields corresponded to the markup of its template from the *templates* section of the MIDV-2020 dataset. To perform manipulations with personal data, information about the position of requisites on the document was taken from the template markup. Additionally, during the creation of each manipulated image, a binary mask was created, where the pixels defining the modified pixels were non-zero.

After performing the specified manipulation, the modified document image was transformed back into the original coordinates. The manipulation mask underwent a similar transformation.

The final manipulated image for all types, except for gluing (see Fig. 5) and overlaying foreign objects (see Fig. 10), was obtained by blending the modified image with the original one, where the values from the processed binary manipulation mask were used as the blending coefficients. The processing of the mask involved ensuring that pixels with a value of zero remained unchanged, while pixels with non-zero values were assigned values from the range  $[0, 1]$ , decreasing as they moved from the interior of the manipulated regions toward their boundaries. To achieve this, the mask was sequentially subjected to erosion and Gaussian blurring, followed by setting the values to zero in pixels that were zero in the original mask.

When creating manipulations involving two images, a random photograph of a document with the same type as the document in the base image was used as the donor image.

Next, the details of creating specific manipulations will be described in detail.

### 2.1. Copy-move manipulation of characters

To obtain a rectangle defining the region of an individual character, we sequentially applied binarization and connected component analysis. Next, we selected a random pair of characters with similar sizes from the found components. The rectangular region of the second character was scaled to match the dimensions of the rectangular region of the first character. Then, linear color correction was performed, after which the minimum and maximum values of each channel in the scaled region became equal to the corresponding values in the first region. The color-corrected scaled region was inserted in place of the first region.

Examples of this type of manipulation are shown in Fig. 1.

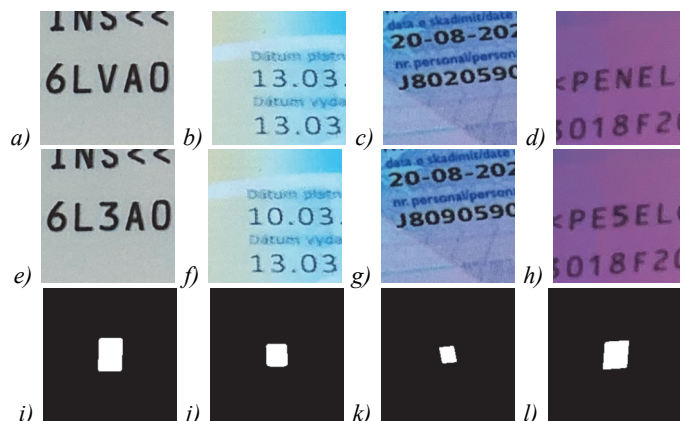


Fig. 1. Examples of copy-move of symbols from MIDV-DM: (a)-(d) – genuine images, (e)-(h) – corresponding forged images, (i)-(l) – corresponding masks indicating the area of manipulation

### 2.2 Splicing of text fields, photo regions, and the entire document

To create splicing-type manipulations, we selected pairs of corresponding regions from the base and donor images. For document splicing, these were always document regions; for splicing in the photo region, the regions were either photos or faces; and for splicing text fields, pairs of corresponding text requisites were used. The region selected from the donor image was inserted into the region selected from the base image, with color correction as described in Section 2.1.

Examples of this type of manipulation are shown in Fig. 2 for individual fields, Fig. 3 for photos, and Fig. 4 for the entire document.

### 2.3. Gluing

In the cases of fraud we analyzed, the merging of images was performed along a horizontal or vertical line, with background mismatches being unnoticeable since the document images were cropped. In our case, the background occupies a significant portion of the image, making it impossible to merge entire images without losing plausibility. Moreover, unlike real fraud cases, the documents in our images are not geometrically normalized, which complicates merging along horizontal and vertical lines, as this could lead to intersections with dynamic fields and cause significant mismatches. Instead, we implemented merging only within the document area, along a line that is vertical or horizontal

in the document's coordinates but not in the image's coordinates. The merging line was randomly selected along one side of a random dynamic field with a small offset.

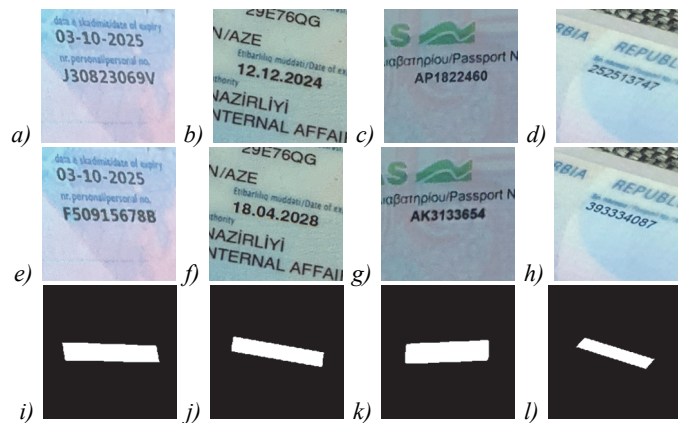


Fig. 2. Examples of text fields splicing from MIDV-DM: (a)–(d) – genuine images, (e)–(h) – corresponding forged images, (i)–(l) – corresponding masks indicating the area of manipulation

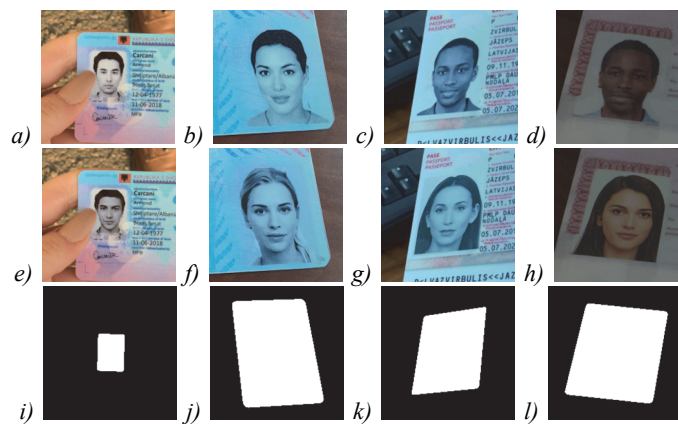


Fig. 3. Examples of photo splicing from MIDV-DM: (a)–(d) – genuine images, (e)–(h) – corresponding forged images, (i)–(l) – corresponding masks indicating the area of manipulation



Fig. 4. Examples of document splicing from MIDV-DM: (a)–(d) – genuine images, (e)–(h) – corresponding forged images, (i)–(l) – corresponding masks indicating the area of manipulation

Examples of this type of manipulation are shown in Fig. 5.

#### 2.4. Overlaying printed text

For the manipulation of overlaying printed text, we either randomly selected a single text field or processed all text fields. During the processing of text fields, we first masked them using one of the methods described in Section 2.5.

To create the overlaid text image, we used the ImageMagick7 utility and one randomly selected font from ten fonts distributed under the SIL Open Font License 1.1. The resulting text image was then scaled to the size of the processed text field and overlaid onto it using the ImageMagick7 *composite* command.

Examples of this type of manipulation for a single field are shown in Figs. 6a and b, and for all fields — in Figs. 6c and d.



Fig. 5. Examples of gluing from MIDV-DM: (a)–(d) – genuine images, (e)–(h) – corresponding forged images, (i)–(l) – corresponding masks indicating the area of manipulation

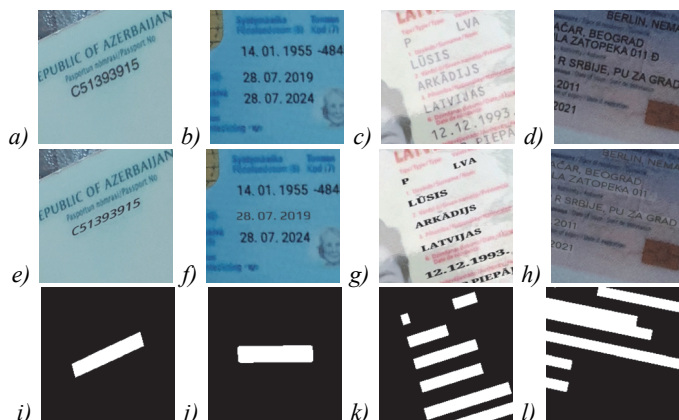


Fig. 6. Examples of text overlay from MIDV-DM: (a)–(d) – genuine images, (e)–(h) – corresponding forged images, (i)–(l) – corresponding masks indicating the area of manipulation

2.5 Masking of information

To implement this type of manipulation, we randomly selected a field on the document or the document itself. Then, we modified the contents of the selected area using one of the following methods:

- Gaussian blur or mean blur;
- Reduction of resolution;
- Dilation;
- Random pixel shuffling, as shown in Fig. 7a;
- Filling with a random color or the average color within the masked area, as shown in Fig. 7b;
- Filling with the first pixel row or column, as shown in Fig. 7c;
- Filling with fragments from the boundary of the region, as shown in Fig. 7d.

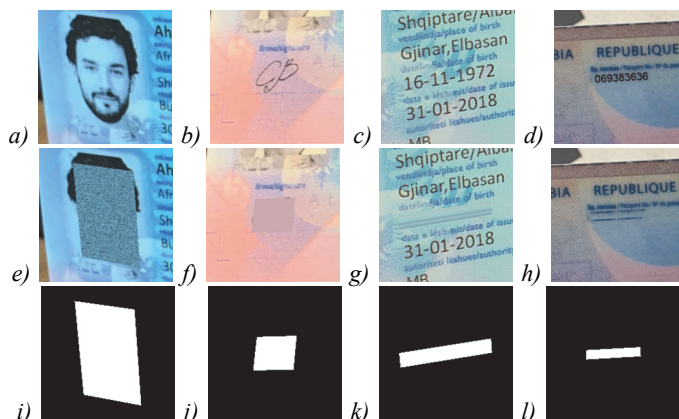


Fig. 7. Examples of masking from MIDV-DM: (a)–(d) – genuine images, (e)–(h) – corresponding forged images, (i)–(l) – corresponding masks indicating the area of manipulation

### 2.6. Overlaying foreign objects

To simulate this type of manipulation, we used two types of objects: text labels and vector graphic elements. For text images, a list of expressions typical for watermarks was compiled. Text images were generated using various colors, slants, and ten fonts under the SIL Open Font License 1.1. The overlay of text images was performed according to three scenarios: a single overlay on the document image without transparency (see examples in Figs. 8*a* and *b*), a single overlay in one of five document positions (corners and center) with varying transparency (see examples in Figs. 8*c* and *d*), and a periodic overlay with random transparency (see examples in Fig. 9). The document image was geometrically normalized only in the first scenario (text overlay without transparency), while the text in one of the five positions and the periodic text were overlaid on the original image.

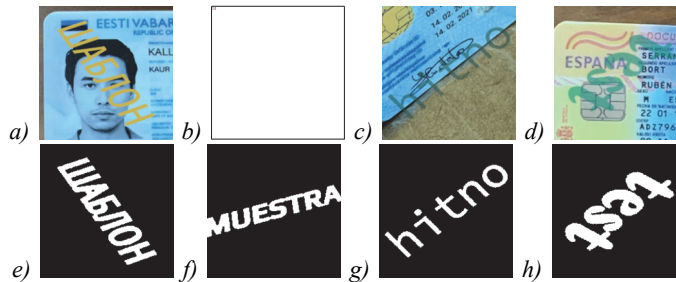


Fig. 8. Examples of foreign individual text overlaying from MIDV-DM: (e)–(h) – forged images, (i)–(l) – corresponding masks indicating the area of manipulation

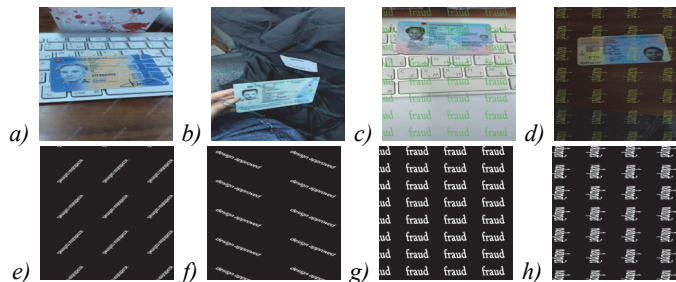


Fig. 9. Examples of foreign periodic text overlaying from MIDV-DM: (e)–(h) – forged images, (i)–(l) – corresponding masks indicating the area of manipulation

The overlay of vector graphic elements included three visualization options, which were randomly selected: outlining fields (see examples in Figs. 10*a* and *b*), pointing with an arrow (see example in Fig. 10*c*), and underlining (see example in Fig. 10*d*) in various orientations. The number of elements, their width, and color (blue, green, or red) were determined randomly.

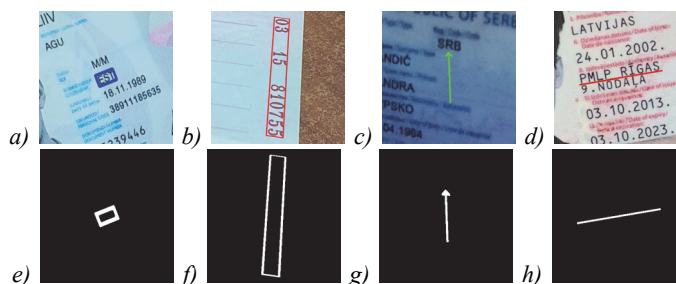


Fig. 10. Examples of vector graphic elements overlaying from MIDV-DM: (e)–(h) – forged images, (i)–(l) – corresponding masks indicating the area of manipulation

### 3. Dataset structure

The root folder of the dataset contains three main directories: *images*, *masks*, and *annotations*. The *images* directory contains nine subdirectories, where the nested directory *authentic* contains the original images from MIDV-2020, while the other eight directories are named according to the manipulation type described in Section 1 and contain images from *authentic* that have undergone the corresponding manipulation. Each of the nine directories contains 10 subdirectories, whose names match the document type identifiers from MIDV-2020 and include 100 images of various documents of the corresponding type. The image names match the names of the base images.

The *masks* and *annotations* directories have a similar structure and contain binary masks defining the manipulation area and annotations in JSON format. The annotations include information about the base image, the type of manipulation performed on the base image, and, if the manipulation involves a donor image, information about the donor image.

#### 4. Benchmark

The IMDL task involves detecting the fact of manipulation and accurately localizing the areas on the images that were modified during the manipulation. This task includes two main aspects:

*Detection.* Determining whether the image has been manipulated. This is a binary classification at the image level.

*Localization.* Precisely identifying the areas on the image that have been altered. This is a binary classification at the pixel level.

Thus, the output of an algorithm solving the IMDL task is a binary response for the image and a binary image with the same dimensions as the input image, indicating the modified pixels. To evaluate the performance of such an algorithm on MIDV-DM, we will use the quality metrics False Positive Rate (FPR), Recall, Precision, and F1-score.

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

In the context of these metrics, we will consider that images and pixels subjected to manipulation belong to the *positive* class.

To assess how state-of-the-art IMDL methods perform on the proposed dataset, we obtained quality metrics for the IML-ViT model [16] on MIDV-DM. Additionally, to evaluate the gap between IMDL methods in the domain of document images compared to the domain of natural images, we compared the F1-score values in the localization task obtained on MIDV-DM and the classic IMDL dataset CASIAv1. The IML-ViT model was chosen because, based on the experimental results from [5], it achieves the highest average F1-score in the localization task under the Protocol-CAT evaluation protocol among other known models in the IMDL field.

To reproduce it, we used the official repository from [16] and the checkpoint named *iml-vit\_checkpoint\_trufor\_20231104*. IML-ViT takes an image with dimensions  $1024 \times 1024 \times 3$ , and its output is an image with dimensions  $1024 \times 1024 \times 1$ , with values ranging from 0 to 1. To meet the input requirements, we first padded all images from MIDV-DM with zeros to make their linear dimensions multiples of 1024 and then split them into non-overlapping patches of size  $1024 \times 1024$ . Since the IML-ViT model does not provide a direct output for the detection task, we consider an image as detected as manipulated if at least one pixel is detected as manipulated in the localization result. We investigated the model's performance using the pre-suggested binarization threshold of 0.5 (under fully blind conditions) and with a threshold tuned on MIDV-DM. The best threshold for F1-score in the localization task was found to be 0.961.

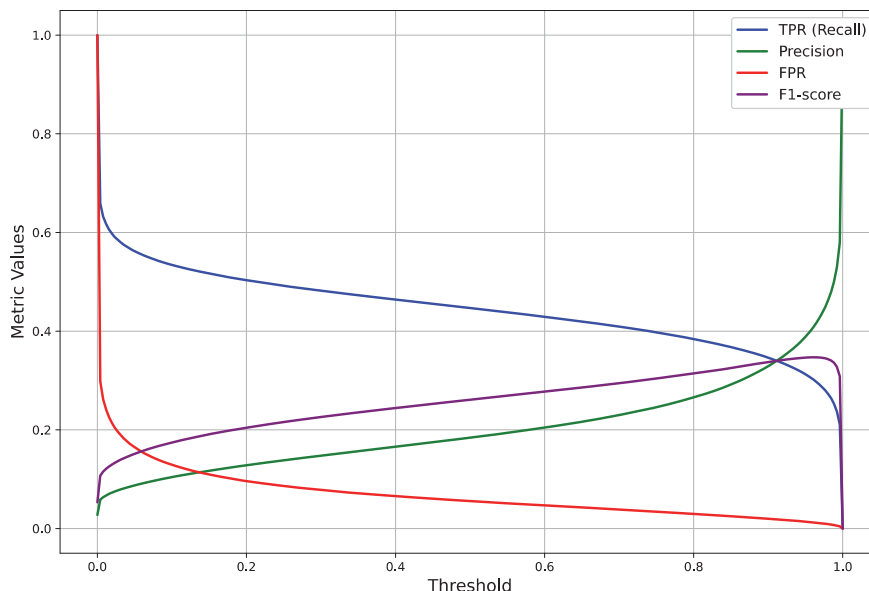


Fig. 11. Values of the IML-ViT model quality metrics at the Pixel-Level, depending on the threshold

The results of IML-ViT are presented in Tables 1, 2. The results for the CASIAv1 dataset [10] are taken from [5].

Tab. 1. Results of IML-ViT (Image-Level)

Method	FPR	Recall	Precision	F1-score
CASIAv1 [10]	–	–	–	–
MIDV-DM (thr = 0.5)	1.0	1.0	0.889	0.941
MIDV-DM (thr = 0.961)	0.987	0.991	0.889	0.937

Tab. 2. Results of IML-ViT (Pixel-Level)

Method	FPR	Recall	Precision	F1-score
CASIAv1 [10]	–	–	–	0.795
MIDV-DM (thr = 0.5)	0.056	0.447	0.184	0.261
MIDV-DM (thr = 0.961)	0.012	0.302	0.409	<b>0.347</b>

As can be seen from Tab. 1, 2, IML-ViT achieves an F1-score of 0.261 in the localization task on MIDV-DM with a fixed threshold and 0.347 with a tuned threshold, which falls significantly short of the best metric values. In the detection task, almost all images were classified as manipulated, with an FPR of 0.987 and a Recall of 0.991 when using the tuned threshold. Additionally, IML-ViT performs worse on MIDV-DM compared to CASIAv1, where the F1-score in the localization task is 0.795. We see this as potential for the development of IMDL methods in the domain of document images. We expect that future approaches will gradually achieve performance comparable to IMDL methods on natural image datasets. MIDV-DM may be extended with new types of manipulations as they appear in real-world scenarios.

### Conclusion

In this work, we propose a new benchmark, MIDV-DM, which includes a dataset of 1000 identity document images not subjected to any software manipulation and 8000 images subjected to one of eight different types of manipulations. In addition to manipulations with areas containing the textual requisites of the document holder, MIDV-DM includes manipulations with areas containing the signature, the holder's photo, an arbitrary part of the document, or the entire document. Additionally, MIDV-DM contains document images reflecting interventions typical of images taken from the Internet: the presence of watermarks and vector graphics elements, as well as the masking of part of the information. The work also presents the quality metrics of the state-of-the-art IML-ViT model on the proposed dataset. These results demonstrate the potential for advancing image manipulation detection and localization methods in the context of document images. The proposed dataset will be useful for advancing research in the field of document image authenticity verification.

### References

- [1] Matalov D P, Usilin S A, Arlazarov V V. Modification of the Viola-Jones approach for the detection of the government seal stamp of the Russian Federation. In: Proc Int Conf Machine Vision (ICMV) 2018; SPIE; 2019. Vol. 11041: 110411Y-1-110411Y-7. DOI: 10.1117/12.2522793.
- [2] Polevoy D V, Panfilova E I, Nikolaev D P. White balance correction for detection of holograms in color images of black and white photographs [In Russian]. Inform Techn Comput Syst 2021; (3): 82-95. DOI: 10.14357/20718632210308.
- [3] Kada O, Kurtz C, van Kieu C, Vincent N. Hologram detection for identity document authentication. In: El Yacoubi M, Granger E, Yuen P C, Pal U, Vincent N, eds. Pattern Recognition and Artificial Intelligence. Cham: Springer; 2022: 346-357. DOI: 10.1007/978-3-031-09037-0\_29.
- [4] Chernyshova Y S, Aliev M A, Sheshkus A V. Optical font recognition of images captured with mobile devices and its application for detecting identity documents forgery [In Russian]. Trudy ISA RAN 2018; 68(S1): 183-191. DOI: 10.14357/20790279180521.
- [5] Ma X, Zhu X, Su L, Du B, Jiang Z, Tong B, et al. IMDL-BenCo: A comprehensive benchmark and codebase for image manipulation detection & localization. arXiv preprint arXiv:2406.10580 (2024).
- [6] Verma M, Singh D. Survey on image copy-move forgery detection. Multimed Tools Appl 2024; 83: 23761-23797. DOI: 10.1007/s11042-023-16455-x.
- [7] Barglazan A-A, Brad R, Constantinescu C. Image inpainting forgery detection: A review. J Imaging 2024; 10(2): 42. DOI: 10.3390/jimaging10020042.
- [8] Tralic D, Zupancic I, Grgic S, Grgic M. CoMoFoD — new database for copy-move forgery detection. In: Proc ELMAR; 2013: 49-54.
- [9] Amerini I, Ballan L, Caldelli R, Del Bimbo A, Serra G. A SIFT-based forensic method for copy-move attack detection and transformation recovery. IEEE Trans Inf Forensics Secur 2011; 6(3): 1099-1110. DOI: 10.1109/TIFS.2011.2129512.
- [10] Dong J, Wang W, Tan T. CASIA image tampering detection evaluation database. In: Proc IEEE China Summit & Int Conf Signal Inf Process; 2013: 422-426. DOI: 10.1109/ChinaSIP.2013.6625374.
- [11] Wen B, Zhu Y, Subramanian R, Ng T-T, Shen X, Winkler S. COVERAGE — a novel database for copy-move forgery detection. In: Proc IEEE Int Conf Image Process (ICIP); 2016: 161-165. DOI: 10.1109/ICIP.2016.7532339.
- [12] Mahfoudi G, Morain-Nicolier F, Retraint F, Pic M. CMID: A new dataset for copy-move forgeries on ID documents. In: Proc IEEE Int Conf Image Process (ICIP); 2021: 3028-3032. DOI: 10.1109/ICIP42928.2021.9506723.
- [13] Sidere N, Cruz F, Coustaty M, Ogier J-M. A dataset for forgery detection and spotting in document images. In: Proc Int Conf Emerging Security Technologies (EST); 2017: 26-31. DOI: 10.1109/EST.2017.8090394.
- [14] Al-Ghadi M, Ming Z, Gomez-Krämer P, Burie J-C, Coustaty M, Sidere N. Guilloche detection for ID authentication: A dataset and baselines. In: Proc IEEE Int Workshop Multimedia Signal Process (MMSP); 2023: 1-6. DOI: 10.1109/MMSP59012.2023.10337681.
- [15] Bulatov K B, Emelyanova E V, Tropin D V, Skoryukina N S, Chernyshova Y S, Sheshkus A V, et al. MIDV-2020: A comprehensive benchmark dataset for identity document analysis. Comput Opt 2022; 46(2): 252-270. DOI: 10.18287/2412-6179-CO-1006.
- [16] Ma X, Du B, Jiang Z, Du X, Al Hammadi A Y, Zhou J. IML-ViT: Benchmarking image manipulation localization by vision transformer. arXiv preprint arXiv:2307.14863 (2024).

---

### *Authors' information*

**Aleksandr Chuiko** (b. 1999) received specialist degree in fundamental mathematics and mechanics from the Lomonosov Moscow State University in 2023. He is currently a PhD student at the Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences. His research interests include image analysis, computer vision, machine learning. E-mail: [a.chuyko@smartengines.com](mailto:a.chuyko@smartengines.com)

**Irina Kunina** (b. 1992) graduated from Moscow Institute of Physics and Technology (State University) in 2015. Ph.D in Technical Science, is a researcher at the Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences. Research interests are image analysis, computer vision, camera calibration. E-mail: [i.kunina@smartengines.com](mailto:i.kunina@smartengines.com)

**Sergey Usilin** (b. 1986) graduated from Moscow Institute of Physics and Technology in 2009. Received his Ph.D. degree in 2018. Currently he works as a lead researcher at the Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences. Since 2016, he has been a executive director of Smart Engines Service LLC. Scope of scientific interests: object detection, machine learning, recognition systems, digital image processing. E-mail: [usilin@smartengines.com](mailto:usilin@smartengines.com)

**Changsheng Chen** (b. 1986) received the B.E. degree in software engineering from Sun Yat-sen University, Guangzhou, China, in 2008, and the Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2013. From 2013 to 2015, he was a Post-Doctoral Research Associate with the Department of Electronic and Computer Engineering at The Hong Kong University of Science and Technology, Hong Kong. Since May 2025, he has been a Full Professor with the Faculty of Engineering, Shenzhen MSU-BIT University. His current research interests include multimedia forensics and security, document understanding, pattern recognition, and machine learning. E-mail: [cschen@smbu.edu.cn](mailto:cschen@smbu.edu.cn)

**Shunquan Tan** (b. 1980) received the B.S. degree in computational mathematics and applied software and the Ph.D. degree in computer software and theory from Sun Yat-sen University, Guangzhou, China, in 2002 and 2007, respectively. He was a Visiting Scholar with New Jersey Institute of Technology, Newark, NJ, USA, from 2005 to 2006. He is currently a Professor with the Guangdong Laboratory of Machine Perception and Intelligent Computing, Faculty of Engineering, Shenzhen MSU-BIT University, Shenzhen, China. His current research interests include multimedia security, multimedia forensics, and machine learning. E-mail: [tansq@smbu.edu.cn](mailto:tansq@smbu.edu.cn)

**Dmitry Nikolaev** (b. 1978) was awarded a Doctor of Sciences in Computer Science in 2023. Since 2007, he has been the Head of the Vision Systems Laboratory, Institute for Information Transmission Problems, Russian Academy of Sciences, Moscow, and he has been the CTO of Smart Engines Service LLC, Moscow, since 2016. Since 2016, he has been an Associate Professor with the Moscow Institute of Physics and Technology (State University), Moscow, teaching the Image Processing and Analysis Course. Now he is the Head of the Vision Systems Laboratory at Federal Research Center “Computer Science and Control” of the Russian Academy of Sciences. He has authored over 150 Scopus-indexed publications and 15 patents. His research interests encompass computer vision and color image understanding. E-mail: [d.p.nikolaev@smartengines.com](mailto:d.p.nikolaev@smartengines.com)

**Vladimir Arlazarov** (b. 1976) received the Specialist degree in applied mathematics from the Moscow Institute of Steel and Alloys, in 1999, the Ph.D. degree in computer science, in 2005, and the Doctor of technical sciences degree in 2023. Since 1999, he has been a senior researcher at the Federal Research Center Computer Science and Control, Russian Academy of Sciences. Since 2016, he is the General Director of the Smart Engines Service LLC. He has published over 150 articles and authored 9 United States patents. His research interests include computer vision and document analysis systems. E-mail: [vva@smartengines.com](mailto:vva@smartengines.com)

---

*Received: July02, 2025. Final version – October 08, 2025.*

---