An End-to-End Computer Vision System for Structured Information Extraction from Turkish ID Card Images

Seyf Kazamel[1(✉)], Umut Kocasarı[2], and Ali Alıcı[1]

[1] Armut Teknoloji AŞ, Üsküdar 34660, Istanbul, Türkiye
{mohammedk,ali}@armut.com
[2] Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany
umut.kocasar@tum.de

Abstract. This paper introduces an end-to-end computer vision pipeline for automatically extracting key personal information from Turkish ID card images. The system addresses real-world challenges such as invalid images, incorrect card orientation, and older ID card formats by integrating advanced methods for card detection, card format classification, orientation correction, text region segmentation, and optical character recognition (OCR). Specifically, we employ YOLOv5 for ID card detection, CLIP to validate that the image is an official Turkish ID and to distinguish between old and new ID formats, CRAFT and U-Net to segment relevant textual regions, and Tesseract for final OCR. By interfacing with APIs from the Turkish Population Registry, our system performs near real-time ID validation while significantly reducing manual effort. Experimental results indicate over 90% accuracy in identifying and classifying ID cards, with OCR word-level accuracy surpassing 96%, and an end-to-end card-level accuracy of 73% for the pipeline, all with execution times under five seconds. Deployed in production, the system saves manual labor hours and speeds up processing workflows. Our findings underscore the feasibility and business value of automated ID verification, highlighting how deep learning–driven ID processing can streamline operations, mitigate fraud, and enhance the user experience.

Keywords: Computer Vision · Optical Character Recognition (OCR) · Turkish ID Cards · Identity Verification · YOLOv5 · CLIP · CRAFT · U-Net · Tesseract

1 Introduction

Identity verification frequently relies on extracting key details from ID cards, making it essential for applications in banking, online services, and government administration. Manual transcription of ID information is prone to errors, slow, and difficult to scale. Recent advances in deep learning and optical character recognition (OCR) have made it possible to automate ID card data extraction with high accuracy. This is particularly important for Turkish ID cards, which come in both older laminated and newer electronic formats, necessitating robust classification and processing strategies. In real-world use

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2025
O. Gervasi et al. (Eds.): ICCSA 2025, LNCS 15649, pp. 51–64, 2025.
https://doi.org/10.1007/978-3-031-96997-3_4

cases, ID images may feature poor lighting, glare, arbitrary rotations, and varied backgrounds, all of which demand efficient, resilient computer vision pipelines capable of fast processing.

Automating the extraction of key personal details (e.g., name, birth date, national identity number) from Turkish ID card images is a challenging task, especially when these images are captured under inconsistent conditions such as uneven lighting, varying angles, or cluttered backgrounds. Accuracy is paramount for ensuring legitimate identities, while efficiency is needed to handle large volumes quickly. Just as important are privacy concerns, which require that sensitive personal data remain secure and, ideally, be processed in-house rather than by third-party services.

Because identity cards contain highly sensitive information, a key motivation for developing an internal system is to avoid sending user data to external information extraction services. Keeping the OCR and extraction process on proprietary servers minimizes exposure of ID images to third parties (aside from the minimal data securely shared with the governmental validation service). This approach substantially reduces the risk of data breaches or misuse by third parties. Moreover, removing the need for routine human review enhances privacy further by limiting additional exposure of personal data.

However, building an entirely in-house, end-to-end ID card extraction system poses several research and engineering challenges. First, robust image classification is required to distinguish valid Turkish ID cards from other documents or photos that might be erroneously uploaded. Second, once an ID card is confirmed, the system must detect, crop, and reorient it—ensuring the text regions line up properly for subsequent extraction. Third, old and new Turkish ID formats exhibit differing text layouts, label languages, and background templates, demanding a mechanism to classify each card's format quickly so the correct downstream model or processing path can be applied. Fourth, localizing and extracting text regions accurately is nontrivial given variations in lighting, orientation, surface glare, and possible obstructions (e.g., fingers holding the card). Finally, maintaining high recognition accuracy on critical fields such as name, surname, date of birth, and ID number is vital to preempt service or authentication errors downstream.

In addressing these challenges, we developed an automated pipeline that integrates modern deep learning components for classification, detection, segmentation, and OCR. Our design leverages: (1) CLIP for zero-shot classification to confirm whether the uploaded image is indeed a Turkish ID card, (2) YOLOv5 for fast and robust card localization, (3) further CLIP-based logic to differentiate old from new ID formats, (4) CRAFT and a U-Net model for pinpointing relevant text fields, and (5) Tesseract for the OCR stage. By unifying these methods in a coherent pipeline, we achieve accurate extraction results under realistic conditions. Lastly, the extracted fields are automatically verified through the Kimlik Paylaşımı Sistemi (KPS) to confirm the correctness of the ID number and personal details, a step that is crucial for reducing fraud and typographical errors.

The remainder of this paper is organized as follows.

- Section 2 reviews relevant works in identity-document processing and the key techniques that underpin modern OCR pipelines.
- Section 3 describes the overall architecture of our system, detailing each processing step, including card detection, orientation correction, and text field segmentation.

- Section 4 outlines the experimental setup and provides quantitative results, illustrating the system's efficacy and performance in practice.
- Finally, Sect. 5 draws conclusions and proposes directions for future work, such as improved OCR strategies, support for additional document types, and enhanced resilience to occlusion or low-quality uploads.

In this work, we show that a dedicated end-to-end computer vision solution can reliably extract ID card information while reducing privacy risks and manual effort in high-volume, real-world environments.

2 Related Works

Optical character recognition (OCR) of identity documents has a long history in the broader field of document processing and pattern recognition. Early approaches often relied on rule-based or classical machine learning strategies for character segmentation, combined with template matching or simple feature extraction techniques (Shaaban et al., 2014). In particular, these older methods were primarily constrained by the need for step-by-step character segmentation (e.g., baseline-based) and suffered from reduced accuracy when confronted with variable-quality images or non-Latin scripts. While effective for their time, these methods depend on handcrafted rules and require substantial domain expertise for each new language or ID format.

More recent methods tend to incorporate deep learning, which improves the robustness of text localization, segmentation, and recognition. For instance, convolutional neural networks (CNNs) and end-to-end pipelines have emerged in various works (Maung & Aye, 2023; Bychkov et al., 2023). These pipelines often integrate region proposal networks or fully convolutional segmentation models for text region detection and then apply attention-based sequence learning or recurrent neural network layers for final text extraction. By leveraging large training datasets, these approaches have shown significant gains in accuracy and generalizability, even under challenging conditions such as poor lighting or skewed orientations (Suddul & Seguin 2024).

Furthermore, some studies focus on specialized mobile-based solutions that require lightweight models and fast inference (Harefa et al., 2022). Such systems often refine or adapt deep learning pipelines to fit real-time constraints in edge devices, illustrating the importance of speed-accuracy trade-offs. Others propose integrated frameworks that apply advanced pre-processing (e.g., generative adversarial networks for image enhancement) or post-processing (e.g., domain-specific correction rules) to further refine OCR outputs (Jobi & Varghese 2023).

Within these more modern systems, identity cards and similarly structured documents receive special emphasis, given the need to handle name fields, national ID numbers, birthdates, addresses, and so forth in a highly reliable manner (Bychkov et al., 2023). Accurate extraction and classification of these text fields opens avenues for streamlined e-KYC (electronic Know Your Customer) and secure onboarding in domains such as finance, governmental services, and telecommunications (Suddul & Seguin 2024).

Existing research demonstrates that combining object detectors, segmentation models, and OCR can significantly improve accuracy and speed in ID recognition. Early methods relied on template matching and manual feature extraction, but they often failed

under challenging imaging conditions. Deep learning–based detection models such as YOLO have proven effective for quickly localizing ID cards in cluttered or poorly lit images, while specialized text detectors like CRAFT (Baek et al., 2019) or segmentation models like U-Net (Ronneberger et al., 2015) accurately identify text regions. For OCR, Tesseract (Smith, 2007) remains a strong open-source solution, particularly when text areas are well-segmented. Several studies have validated these approaches: Tavakolian et al. (2020) integrated a deep text detector and face detection for ID card scanning, Cavalcante et al. (2024) combined object detection with OCR for ID badge reading, and Suddul et al. (2024) employed U-Net for text segmentation, all achieving high accuracy in their respective domains.

Building on these insights, our work employs a deep learning pipeline for Turkish ID card images, unifying classification, text region segmentation, and OCR into an efficient end-to-end architecture. We leverage advanced approaches such as object detection, robust segmentation techniques, and open-source pretrained sequence recognition networks while addressing domain-specific challenges like older versus newer card formats. This design focuses on delivering a streamlined, high-performance solution for structured information extraction from Turkish IDs.

3 System Overview

Our end-to-end system is organized as a pipeline of modules, each responsible for a specific task in the ID processing workflow. Figure 1 shows the steps of the pipeline.

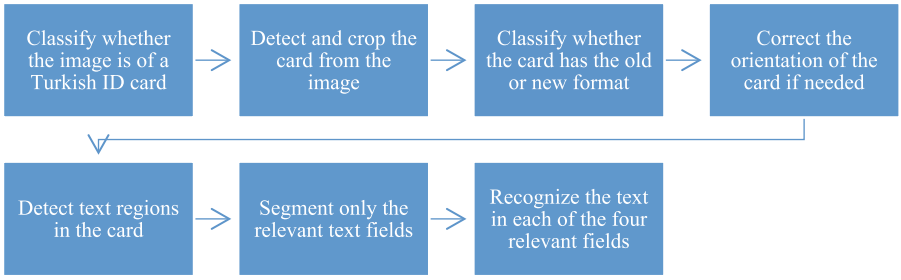

Fig. 1. Information extraction pipeline steps

3.1 Image Classification and Card Preprocessing

The input to our system is an image that is presumed to contain an ID card (often a photo taken by a mobile device). Before we detect and recognize the text on the card, we first check if the photo is of a valid ID card, crop the image to focus on the card, and classify whether the card has the old or new format in order to process it accordingly.

ID Card Identification Using CLIP. We utilize the CLIP model (Radford et al., 2021) in a zero-shot classification manner for ID card identification. We use CLIP's pretrained

image encoder and text encoder where we encode the image, and we encode many candidate texts, e.g., “a photo of a Turkish ID card”, “a photo of a passport”, “a photo of a driving license”, “a photo of a person”, “a photo of a scene”, “a photo of a document”, etc. By comparing the cosine similarity of image embeddings to these text embeddings, we can determine if the image is likely an ID card. If the similarity to the correct “ID card” prompt is significantly higher than to generic prompts, we accept the image; otherwise, we flag it as an invalid input. CLIP’s ability to understand image content in terms of text is remarkable, and it has been shown to implicitly perform OCR and fine-grained classification during its training. We found CLIP to be robust in filtering out images that were not ID cards (for example, it correctly rejected a few test images of driver’s licenses and passports as not being Turkish ID cards, since our prompt was specific).

Card Detection using YOLOv5. Once the image is confirmed to contain an ID card, the next step is to locate the card within the image. Users often upload photos where the ID card is on a table or held in hand against a background. We employ YOLOv5, a one-stage object detector, to detect the bounding box of the ID card. We trained a YOLOv5 model on a custom dataset of ID card images, labeled with bounding boxes over the ID cards. The model treats the ID card as a single object class. YOLOv5 was chosen for its balance of speed and accuracy; it can process images quickly and its smaller models can run on CPU if needed. Compared to two-stage detectors, YOLO’s single-shot approach is faster and sufficiently accurate for our needs.

The output at runtime is the bounding box coordinates of the ID card. We crop the image to this bounding box (with a slight margin) to isolate the card. By reducing the image to just the ID card, we eliminate background noise and accelerate the following processing (which can then operate on a smaller image region) (Fig. 2).



Fig. 2. An illustration of ID card detection in an image (the card in the picture is artificial)

ID Card Format Classification Using CLIP Embeddings. After the image is validated as an ID card, we further classify which type: old vs. new Turkish ID. We do this by again using CLIP. We precalculated the CLIP embeddings of 500 old format and 500

new format ID cards, and at inference time, we compute the embedding of the current card and then calculate its cosine similarity with the reference embeddings. We assign the class of the most similar three embeddings to the current card. This classification is important because the text fields and layout differ. For instance, the old IDs have fields like "Nüfus Cüzdanı" labels, whereas the new IDs have English translations for field labels. Knowing the type allows us to apply appropriate downstream processing.

Orientation Correction and Resizing. The Orientation Correction step is essential for aligning the ID card image before further processing. In this stage, the pipeline utilizes a pre-trained face detection model—specifically, a Single Shot MultiBox Detector with a ResNet-10 backbone—to detect facial features on the card. The process involves rotating the image through four potential orientations (0°, 90°, 180°, and 270°) and evaluating each based on the face detection confidence score. The orientation yielding the highest confidence is selected as the correct alignment, ensuring that any card is properly oriented. After determining the correct orientation, the image is resized to a standard 640×640 dimension, thereby standardizing the input for subsequent text detection and recognition steps (Fig. 3).



Fig. 3. An illustration of the orientation correction process

3.2 Relevant Text Fields Segmentation with CRAFT and U-Net

After obtaining an upright, background-free ID card image, we first detect all text regions using CRAFT and then segment the four key fields (ID number, surname, name, and date of birth) with a U-Net model. We use different U-Net models depending on whether we are dealing with older or newer ID card layouts.

CRAFT-based Detection: We run CRAFT (Character Region Awareness for Text Detection) to detect potential text regions. CRAFT outputs bounding boxes (and corresponding character density maps) for each region it identifies as text. These bounding boxes can be granular—often detecting individual words or lines.

U-Net Field Segmentation: Next, we apply a U-Net model specifically trained to segment the four relevant fields on the ID card. Rather than highlighting all text pixels, this U-Net produces masks that delineate exactly where the ID number, surname, name,

and date of birth should appear. Since the placement of these fields differs between old and new ID layouts, we employ separate U-Net models accordingly.

Combining CRAFT Boxes with U-Net Fields: Once we have the CRAFT bounding boxes and the U-Net's field segmentation masks, we intersect them to isolate only those boxes that fall within the four designated fields. If multiple horizontally aligned bounding boxes belong to the same field, we merge or combine them as needed so that, in the end, each of the four fields is captured by exactly one bounding box. We iteratively merge the closest boxes until only four remain. Finally, we sort these boxes by their vertical or horizontal position to map them consistently to the correct fields (ID number, surname, name, date of birth).

This combined approach leverages CRAFT's strong performance on text detection while using the U-Net segmentation to pinpoint the specific fields of interest. By intersecting the text bounding boxes with the field masks, we ensure we capture the correct regions for each field—even in the presence of background patterns, varying fonts, or different ID card designs (Fig. 4).

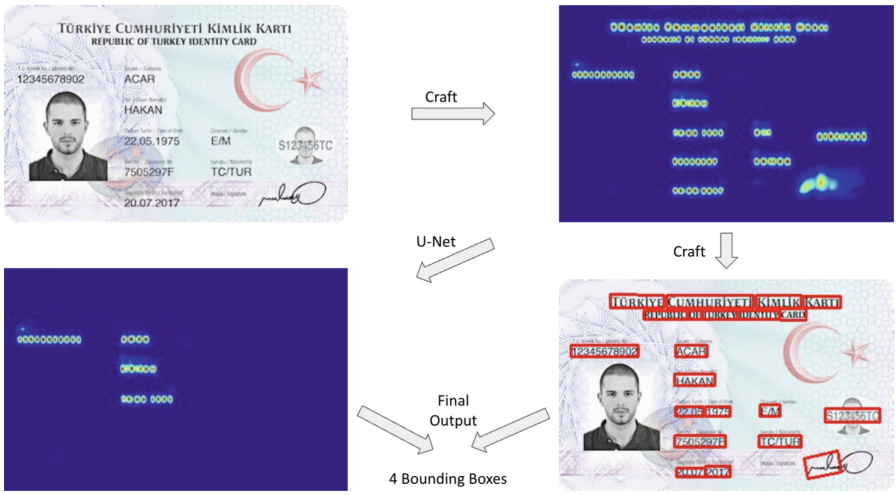

Fig. 4. Using CRAFT to generate the bounding boxes and U-Net segmentation to find the relevant boxes

3.3 Text Extraction Using Tesseract OCR

Once the text regions have been identified, the system extracts the text content from each region using OCR. We utilize Tesseract OCR (version 4.1 with the LSTM engine) configured for both Turkish and English, depending on the field type. Each detected text region (bounding box from the previous step) is cropped from the image and undergoes preprocessing before being fed into Tesseract.

For preprocessing, we apply denoising using `fastNlMeansDenoisingColored` to enhance text clarity. Depending on the field type, we employ different OCR configurations:

- **ID number and birth date:** These fields are processed using the English language model. Additional color thresholding is applied, and multiple Page Segmentation Modes (PSM) are tested, including PSM 3, 6, 7, 8, and 13, to maximize accuracy.
- **Name and surname:** These fields are processed using the Turkish language model. Multiple PSM modes are tested to improve recognition, ensuring accurate extraction of characters such as "Ğ, Ş, İ, Ç, Ö, Ü."

The OCR results undergo post-processing to validate and format the extracted text:

- **ID number:** Must be exactly 11 digits. Any output with non-numeric characters or incorrect length is discarded.
- **Birth date:** Must follow the `DD.MM.YYYY` format, with valid month and day values.
- **Name and surname:** Punctuation is removed, and outputs are checked for expected uppercase formatting in the Turkish alphabet.

To further enhance accuracy, multiple OCR attempts are made with different preprocessing techniques, including adjusting brightness levels and inverting colors. The results are filtered based on contextual rules to eliminate invalid outputs. The output of this stage is a structured set of textual entries extracted from the ID card, such as Name = "HAKAN", Surname = "ACAR", DOB = "22.05.1975", and ID No = "12345678902". These results are then passed to the validation step to ensure validity before further processing (Fig. 5).



Fig. 5. Using Tesseract OCR to recognize the ID information from the cropped boxes

3.4 Data Verification via API

The final stage of the pipeline is to verify the extracted personal information against an authoritative source. After OCR, we have structured data: e.g., {name: "HAKAN", surname: "ACAR", date_of_birth: "22.05.1975", id_number: "12345678902"}. We utilize the Kimlik Paylaşımı Sistemi (KPS) to validate this data. KPS provides a SOAP-based web service that confirms if a given combination of ID number and personal information is valid. Our verification step takes the OCR results and makes a secure SOAP request to the KPS API with the ID number, first and last name, and year of birth. The API responds with true or false on whether the ID info is valid.

If the information matches exactly (e.g., the name and birth year correspond to that ID number in the government database), we mark the extraction as verified. If not, we

flag it: this could mean the OCR made an error (like a misrecognized digit) or that the ID card is not authentic. In case of a mismatch, we leave the process of verifying that card to a human.

This verification step is critical in real deployment scenarios such as automated user onboarding, where just extracting text is not enough and where one also needs to ensure the extracted data is legitimate. By integrating with the KPS API, our system effectively closes the loop from image to validated data. Of course, this step requires access to the government service, which fortunately is available 24/7.

4 Experimental Details and Results

This section summarizes the training setups and performance evaluations of our proposed pipeline for Turkish ID card information extraction. We detail the hyper-parameters and performance metrics for the YOLOv5 detector and the U-Net segmentation models, and we report results for each stage of the pipeline as well as the end-to-end performance.

4.1 YOLOv5 Training and Card Detection

We trained the YOLOv5 model on 1,000 ID card images (500 new, 500 old) using a medium-sized model with an input resolution of 640×640. Data augmentation (mosaic, rotation, horizontal flip, color jitter) was applied to enhance robustness. Key training hyper-parameters are listed in Table 1. During training, we leveraged transfer learning by initializing with YOLOv5 weights pre-trained on COCO and then fine-tuning on our ID card data.

Table 1. YOLOv5 Training Hyper-parameters

Parameter	Value
Optimizer	SGD (momentum = 0.937)
Learning Rate	0.01 (cosine decay to 0.0001)
Batch Size	16
Epochs	100
Augmentation	Mosaic, $\pm$ 5° rotation, flip, jitter

On the test set, YOLOv5 achieved a precision of 96.8%, recall of 95.5%, and a mAP@0.5 of 97.2%, with an average inference time of ~15 ms per image on a GPU (Table 2).

4.2 U-Net Training and Segmentation

For segmenting the four key text fields, two U-Net models (one per ID format) were trained using CRAFT's character score maps. The models used a ResNet-34 encoder

Table 2. YOLOv5 Detection Performance

Metric	Value
Precision	96.8%
Recall	95.5%
mAP@0.5	97.2%
mAP@0.5:0.95	92.5%
Inference Time (GPU)	~15 ms per image

Table 3. U-Net Training Hyper-parameters

Parameter	Value
Architecture	U-Net (ResNet-34 encoder)
Optimizer	Adam
Learning Rate	0.001
Batch Size	8
Epochs	50
Loss Function	BCE + Dice
Augmentation	$\pm 10^{\circ}$ rotation, flip, scaling

optimized with Adam and a combined Binary Cross-Entropy plus Dice loss over 50 epochs. Table 3 lists the hyper-parameters.

The U-Net achieved a pixel accuracy of 98.1%, mean IoU of 90.3%, and Dice coefficient of 94.5%, with an average inference time of ~20 ms per image (Table 4).

Table 4. U-Net Segmentation Performance

Metric	Value
Pixel Accuracy	98.1%
Mean IoU	90.3%
Dice Coefficient	94.5%
Inference Time (GPU)	~20 ms

4.3 Pipeline Steps Evaluation

We evaluated the pipeline steps on 200 Turkish ID card images each. The system integrates CLIP-based card identification, ID photo and format classification, text detection (CRAFT + U-Net), and OCR using Tesseract.

- **ID Card Identification Using CLIP:** The zero-shot classifier achieved 97.5% accuracy. Table 5 shows the confusion matrix for a balanced test set.
- **Overall Pipeline Accuracy:** The end-to-end system achieved a 73% card-level accuracy, defined as the percentage of ID cards for which all four fields were extracted correctly. The main sources of error were occasional OCR misreads and rare missed text field detections under challenging conditions. Card misclassification also contributed to the errors.

Table 5. ID Photo Classification Confusion Matrix

Actual/Predicted	Valid ID Photo	Invalid
Valid ID Photo	98	2
Invalid	3	97

- **ID Format Classification:** The classifier distinguishing new versus old ID formats achieved 95.5% accuracy (Table 6). Minor misclassifications were mostly due to poor lighting conditions.

Table 6. ID Format Classification Confusion Matrix

Actual/Predicted	New ID	Old ID
New ID	95	5
Old ID	4	96

- **ID Card Identification and Classification:** The first three steps of the pipeline, i.e., ID card identification with CLIP, card detection and cropping with YOLOv5, and ID format classification were also jointly evaluated on a dataset of 200 images and yielded an accuracy of 91%.
- **Text Field Detection:** Combining CRAFT and U-Net, the system correctly detected all four expected text fields in 94% of cases. Misdetections were primarily due to low contrast or occlusion in a small subset of images.
- **OCR Performance:** The OCR module yielded a character-level accuracy of 97.5%, a word-level accuracy of 92.0%, and a card-level accuracy of 80% (Table 7). Card-level accuracy is a metric that we introduced because it captures the performance on the ID card information extraction task better than the metrics in the literature. Card level accuracy is the percentage of cards whose information (the four relevant fields) were extracted completely correctly. It is a stringent metric because misrecognizing only one character in any field of the card yields the prediction for the whole card erroneous.

Table 7. OCR Accuracy

Metric	Value
Character-level Accuracy	99.1%
Word-level Accuracy	96.2%
Card-level Accuracy	80.5%

The experimental results demonstrate that our system achieves high accuracy across all processing stages. YOLOv5 robustly detects ID cards, while the U-Net models accurately segment key text regions. The high performance of the classification modules ensures that only valid, correctly oriented images are processed further, thereby enhancing overall system reliability. Although OCR performance is strong, minor errors in low-quality images suggest avenues for future work, such as improved preprocessing and advanced recognition methods. With an overall 73% end-to-end accuracy, the pipeline significantly reduces manual verification effort and underscores the feasibility of automated ID verification in real-world applications.

This result is comparable to findings reported in the literature, such as those by Chandra et al. (2021), who achieved an overall accuracy of 74.2%. It is important to note that their definition of accuracy encompasses the correct prediction of only the full name and the ID card number, whereas our definition also includes the correct prediction of the date of birth.

5 Conclusion and Future Work

We presented an end-to-end computer vision system for extracting and verifying information from Turkish ID card images. The system combines multiple deep learning components in a unified pipeline: YOLOv5 for detecting the ID card in images, CLIP for image validation and classification, CRAFT and U-Net for segmenting relevant text regions, and Tesseract OCR for reading the text. Through a series of preprocessing (cropping, denoising, orientation correction) and postprocessing (OCR result verification and correction), the system achieves a high level of accuracy. It successfully automates what has traditionally been a manual task, handling ID card images in various orientations and conditions and producing structured digital data. Our experimental evaluation showed over 90% accuracy in identifying and classifying ID cards and around 80% card-level accuracy in text extraction, all within an inference time of under 5 s per image. Integrating the Turkish Population Registry API for verification adds an extra layer of reliability, ensuring that the extracted details correspond to a real person.

This study demonstrates the feasibility of a deep learning-driven ID processing system in real-world scenarios. It also illustrates the advantage of combining multiple AI models—each specialized for a sub-problem—into one coherent solution. By leveraging state-of-the-art models (object detectors, image-language models, text detectors, etc.) and existing OCR technology, we were able to build a system relatively quickly that performs competitively.

Future Work: There are several avenues for improvement and extension:

- **Enhanced OCR with Deep Learning:** Replacing or supplementing Tesseract with a modern OCR model (such as CRNN or Transformer-based text recognizer fine-tuned on ID card fonts) could improve the text extraction accuracy. This could be especially beneficial for challenging cases like faint text or uncommon names. It would require assembling a larger training set of annotated ID card text or using synthetic data to train the OCR model.
- **Multi-language and Document Support:** Although we focused on Turkish IDs, the system can be extended to other identity documents (passports, driver's licenses, IDs from other countries). This would involve updating the CLIP prompts, retraining detection on new document types, and using language-specific OCR models. The modular design makes it feasible to maintain one pipeline that handles multiple document types. For instance, the system could first classify the document type (passport vs. ID vs. license) and then route it to the appropriate OCR settings or verification API.
- **Robustness to Occlusion and Damage:** In real usage, an ID card might be partially occluded (e.g., a finger covering a corner) or damaged (faded text, scratches). Future work could incorporate techniques to handle these. For occlusion, one idea is to use multiple frames (if it's a live capture from a video, maybe take the best parts from several frames). For damage, training the text segmentation and OCR on augmented data (with occlusion masks or noise) could help the models become more robust.
- **User Feedback Loop:** In a real application, any time the system fails (for example, cannot verify an ID automatically and requires manual review), that case can be flagged and later added to the training data (active learning). Over time, this can improve the system. Particularly for fields that caused OCR errors, collecting those and retraining a model (or updating a language model for autocorrection) could incrementally push accuracy higher.

In conclusion, our end-to-end system shows that it is practical to automate ID card reading with high accuracy using current computer vision and deep learning techniques. This has significant implications for industries that require identity verification, offering a faster and potentially more secure alternative to manual checking. By continuing to refine the models and expanding the system's capabilities, we move closer to fully automated, reliable ID verification in real-world applications.

Disclosure of Interests. The authors Seyf Kazamel and Ali Alıcı are employed by Armut Teknoloji. Umut Kocasarı was also employed by Armut Teknoloji while implementing this project.

Data Privacy Statement. All processing of private data (ID card images) is performed exclusively on our secure servers which are access-controlled, with no unauthorized external ingress/egress permitted. We ensure that no personally identifiable information is transmitted outside our system and that all data handling complies with applicable privacy regulations and internal security standards.

References

Tavakolian, N., Nazemi, A., Fitzpatrick, D.: Real-time information retrieval from Identity cards. http://arxiv.org/abs/2003.12103 (2020)

Cavalcante, W., Torné, I.G., Camelo, L., Fernandes, R., Printes, A., Bragança, H.: An ID badge information extractor based on object detection and optical character recognition. IEEE Access **12**, 152559–152567 (2024)

Radford, A., et al.: Learning transferable visual models from natural language supervision. In: Proceedings of the 38th International Conference on Machine Learning. pp. 8748–8763. PMLR (2021)

Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9357–9366 (2019)

Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 234–241. Springer International Publishing, Cham (2015)

Shaaban, A., Hussein, A., Shokry, E., Alaa-Eldin, O., Rashwan, M., Barhamtoshy, H.: ID card recognition based on ArabicOCR system. Egypt. J. Lang. Eng. **1**(2), 35–49 (2014)

Smith, R.: An overview of the tesseract OCR engine. In: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), pp. 629–633 (2007)

Bychkov, O., Zubyk, L., Gololobov, D., Isaienkov, Y., Grynkevych, G., Ivanytska, A.: The system for recognizing useful information of the client's ID-card based on machine learning technologies. In: DSMSI (2023)

Jobi, T.R., Varghese, L.: A novel approach for data retrieval from identity cards using deep learning. In: Proceedings of the National Conference on Emerging Computer Applications (NCECA), vol. 5(1), pp. 617–620 (2023)

Suddul, G., Seguin, J.F.L.: A custom-built deep learning approach for text extraction from identity card images. IJ-ICT **13**, 34 (2024)

Maung, S.Z.M., Aye, N.: Text region localization and recognition for ID card identification using deep learning approaches. In: 2023 IEEE Conference on Computer Applications (ICCA), pp. 411–416 (2023)

Harefa, J., Alexander, Chowanda, A., Haikal, E., Fedrick, Antonio Wiranata, S.: ID card storage system using optical character recognition (OCR) on android-based smartphone. In: 2022 International Conference on Electrical and Information Technology (IEIT), pp. 75–79 (2022)

Chandra, A., Stefanus, R.: An end-to-end optical character recognition pipeline for Indonesian identity card. In: 2021 9th International Conference on Information and Communication Technology (ICoICT), pp. 307–312 (2021)