

Automatic TV Screen Blackout in Real Estate Images Using YOLO26 Segmentation

Hossein Khodabakhshi Rafsanjani*
School of Computing
The University of Kent
Canterbury, United Kingdom
H.Khodabakhshi-Rafsanjani@kent.ac.uk

Marek Grzes
School of Computing
The University of Kent
Canterbury, United Kingdom
M.Grzes@kent.ac.uk

Mark Orme
Department of IT
NicheCom
Wokingham, United Kingdom
mark.orme@nichecom.co.uk

Nasibeh Naseri
Independent Researcher
Reading, United Kingdom
nazanin.nn48@gmail.com

Abstract—Automatic TV screen blackout is a labour-intensive post-processing task in real estate photography. This paper proposes an efficient and accurate pipeline for TV screen segmentation and realistic blackout. A comprehensive dataset containing challenging real-world cases such as strong reflections, occlusions and varying lighting conditions is first created. The YOLO26 model is then trained for instance segmentation, after which contour extraction and polygon approximation refine the mask boundaries and remove jagged artifacts. Finally, a mask blending stage applies black filling, linear gradient shading and edge feathering to produce natural-looking blackout results. Quantitative results for precision, recall, mAP and inference time, together with qualitative evaluations, confirm the effectiveness of the proposed method. It successfully addresses key difficulties in real estate images and offers a practical solution for automated TV screen blackout in large-scale photo processing pipelines.

Keywords—Computer Vision, Segmentation, Automation, Deep Learning, YOLO.

I. INTRODUCTION

High-quality images play a vital role in real estate marketing. Professionally edited photographs not only attract greater attention from potential buyers but also contribute to faster sales and higher listing prices [1]. In professional editing workflows, interior photographs frequently contain television screens that must be carefully masked and blacked out to achieve a clean, premium appearance. TV screen blackout is essential to eliminate visual distractions, maintain a professional look, and comply with major listing platform guidelines. This manual process is one of the most repetitive and time-consuming tasks, requiring editors to precisely trace the boundaries of each TV screen and apply black filling with gradient shading to ensure natural integration with the surrounding scene.

Despite its importance, TV screen blackout remains a highly challenging task, not only for manual editors but also for most deep learning-based segmentation models. The process is complicated by several real-world complexities commonly found in real estate interiors, including strong reflections on glossy TV screens, partial occlusions by furniture or decorative items, mirroring of surrounding room elements, and highly variable indoor lighting conditions. These factors frequently cause inaccurate mask boundaries, and leakage into adjacent areas when using general-purpose models. Because automatic TV screen blackout is a commercially valuable task in the real estate photography industry, and to the best of our knowledge there is currently

no dedicated open-source model specifically designed for this purpose, the need for a robust and specialized automated solution has become increasingly critical for large-scale photo processing workflows.

The first and most critical step in automatic TV screen blackout is accurate instance segmentation of the TV screen region. Over the past decade, significant progress has been made in instance segmentation, with notable contributions from Mask R-CNN [2], [3], its variants [4], U-Net [5], and the YOLO series including YOLOv3 [6], YOLOv4 [7], YOLOv5 [8], YOLOv8 [9], YOLOv10 [10], YOLO11 [11], and the very recent YOLO26 [12]. In parallel, transformer-based and foundation models such as the Segment Anything Model (SAM) [13], SAM2 [14], and SAM3 [15] have demonstrated remarkable zero-shot segmentation capabilities. Other prominent approaches include Mask2Former [16], Segmenter [17], and DETR-based methods [18], [19]. Among these, YOLO26 and SAM3 currently stand out as the most recent and powerful models, offering state-of-the-art performance in real-time and high-accuracy segmentation tasks. Due to YOLO26 flexibility, significantly faster inference speed compared to SAM3, and the availability of multiple model scales (nano to extra-large), we selected YOLO26 as the base architecture for our proposed TV screen blackout framework.

Despite the rapid progress in general-purpose instance segmentation, there is currently no dedicated open-source model specifically designed for TV screen segmentation and blackout. This is largely due to the commercial sensitivity of the task within the highly competitive real estate photography industry. To address this gap, a framework is proposed in this paper. First, we construct a comprehensive dataset of real estate interior images containing various challenging conditions, including reflections, occlusions, and diverse lighting. We then train the YOLO26 model for accurate instance segmentation. The segmentation output is further refined using polygon approximation of the extracted contour to preserve the predominant quadrilateral shape of the TV screen while maintaining important concavities caused by occlusions. Finally, a mask blending stage with black filling and linear gradient shading is applied to produce natural-looking blackout results that closely resemble professional manual editing.

The remainder of the paper is organized as follows: Section II provides a detailed description of the proposed method, Section III presents both quantitative and qualitative experimental results, and finally, Section IV concludes the paper.

II. PROPOSED METHOD

Manual editing of TV screens in real estate photographs is a highly labour-intensive process. Editors must carefully trace the boundaries of each TV screen and apply blackout operations, which becomes extremely time-consuming when processing thousands of images per day. To overcome these limitations, we propose an automated approach for TV screen blackout. However, developing an effective automatic solution is particularly challenging because it requires accurate segmentation of TV screens under complex real-world conditions. These segmentation challenges include strong reflections on glossy screens, mirroring of surrounding room elements, partial occlusions by furniture, varying lighting conditions, and the presence of dark rectangular regions that often lead to false positives. Fig. 1 illustrates representative examples of these challenges, including strong reflections and occlusions (Fig. 1(a,b)), dark rectangular objects that frequently cause false positives (Fig. 1(c)), and partial TV screens appearing at the image borders (Fig. 1(d)).

To address these challenges, the proposed method consists of four main components: dataset creation, instance segmentation, post-processing, and final mask application with blending. The YOLO26 model is trained on the custom dataset created in this study and used for accurate detection and pixel-level segmentation of TV screens. The resulting segmentation output then undergoes contour-based post-processing to refine the mask boundaries. Finally, the refined mask is applied to the original image, where edge feathering, black filling, and gradient shading are performed to produce a realistic and artifact-free TV screen appearance suitable for real estate photography.

A. Dataset Creation

A custom dataset comprising 1000 real estate interior photographs are assembled exclusively from our company listings. The images feature TVs captured under realistic and challenging conditions, including reflections from windows or artificial lights, mirroring of surrounding elements, occlusions by stuff, diverse lighting setups, and confounding dark rectangular areas. The images have high resolution (4608×3072) and stored in JPG format.

During the annotation process, the borders of each TV screen are manually marked using polygon points with appropriate class labels. These polygon annotations define a closed boundary that separates the image into two distinct regions: the interior area representing the TV screen (foreground) and the remaining area as background. The annotated data for each image is saved in YOLO format as a text file with the same filename as the corresponding image. Each text file contains the class label along with the normalized x and y coordinates of the polygon vertices that define the object boundaries. The complete dataset is subsequently divided into three subsets: training, validation, and testing sets.

B. Instance Segmentation Model

Performing instance segmentation on TV screens in real estate images presents a significant challenge, and selecting the appropriate model is paramount. The chosen model must demonstrate high accuracy while offering excellent flexibility in terms of speed and hardware compatibility.



Fig.1. Typical challenging images of the dataset.

YOLO26 [12], the latest evolution in the YOLO series, is selected for this work. YOLO26 stands out as one of the most advanced real-time instance segmentation models, delivering an outstanding balance between high precision and fast inference speed. It is particularly well-suited for practical applications due to its streamlined architecture, native end-to-end design, and strong generalization capabilities even with limited training data.

A key strength of YOLO26 is its availability in five different scales: nano (n), small (s), medium (m), large (l), and extra-large (x). This range provides significant flexibility, allowing us to choose the most suitable variant depending on the desired trade-off between accuracy, inference speed, and available computational resources. Smaller variants such as nano and small are ideal for faster inference on edge devices, while larger variants such as large and extra-large can be used when higher accuracy is prioritized.

These characteristics make YOLO26 highly suitable for TV screen segmentation in real estate images. Quantitative results and detailed performance analysis of the selected YOLO26 model are presented in the next section.

C. Post-Processing

Considering the predominantly quadrilateral shape of TV screens in real estate images, the Ramer–Douglas–Peucker algorithm [20] is selected for post-processing with a relatively low approximation tolerance ($\epsilon \approx 0.005 \times \text{perimeter}$). This choice enables the algorithm to straighten minor jagged boundaries introduced by segmentation while keeping each side of the screen close to a straight line in occlusion-free cases. At the same time, it preserves larger irregular concavities and indentations caused by real-world occlusions (e.g., furniture or curtains), which would otherwise be eliminated by a convex hull approach.

The Ramer–Douglas–Peucker algorithm approximates a digital curve represented by an ordered set of points $P = \{p_1, p_2, \dots, p_n\}$ by recursively selecting a subset of vertices that form a simplified polygon. Given a tolerance $\epsilon > 0$, the algorithm ensures that the maximum perpendicular distance from any discarded point to the nearest line segment of the

approximated polygon does not exceed ϵ . Formally, the goal is to find a polyline $Q \subseteq P$ such that:

$$\max_{p_i \in P \setminus Q} d(p_i, Q) \leq \epsilon \quad (1)$$

where $d(p_i, Q)$ denotes the perpendicular distance from point p_i to the closest line segment in the simplified polyline Q . In this work, ϵ is set proportionally to the contour perimeter as $\epsilon = 0.005 \times L$, where L is the arc length (perimeter) of the original contour. Here, the contour is the outer boundary of the detected object.

D. Mask Application and Blending

The refined binary mask is finally applied to the original input image to generate the final output. The mask first undergoes a feathering operation with a radius of 5 pixels to ensure smooth boundary transitions between the TV screen and the background. The interior region of the TV screen is then set to black, and a subtle linear gradient is applied. This blending process is implemented by compositing the feathered mask with a generated linear intensity gradient (ranging from darker to lighter black) over the original image pixels within the masked area. The combination of feathering, black filling, and gradient application produces a clean, realistic appearance of the TV screen that integrates seamlessly into the real estate photograph.

Fig. 2 illustrates the flowchart of the proposed method. The process starts with an input real estate image, which is passed through already trained YOLO26 model for instance segmentation of TV screen. The model outputs a binary mask in which the white areas represent the predicted TV screen regions, as shown in Fig. 3(b). The resulting segmentation output is then processed to extract the outer contour of the detected TV screen. To smooth minor jagged boundaries while preserving important concavities caused by occlusions, the Ramer–Douglas–Peucker algorithm is applied for polygon approximation. Subsequently, the refined mask undergoes feathering, followed by mask blending with the original image. The interior of the TV screen is then filled with black, and a subtle linear gradient is applied to create a natural shading effect.

Fig. 3 presents several intermediate outputs corresponding to key steps in the proposed workflow on a challenging real estate image containing three closely located TV screens with reflections and occlusions, providing a visual understanding of the progressive refinement process. As can be seen in this figure, the trained YOLO26 model produces a reasonable initial segmentation mask, as shown in Fig. 3 (b), but the raw output still contains jagged boundaries as seen in the zoomed-in view in Fig. 3 (c). After applying the Ramer–Douglas–Peucker algorithm, the boundaries are significantly smoother while important concavities caused by occlusions are preserved, as illustrated in Fig. 3 (d). The comparison between the zoomed-in original region in Fig. 3 (e) and the final blackout result in Fig. 3 (g) demonstrates that the combination of accurate segmentation, polygon approximation, and gradient-based blending produces a clean and natural-looking TV screen blackout that closely resembles professional manual editing. More comprehensive results, including quantitative metrics, training curves, and qualitative comparison will be presented in the next section.

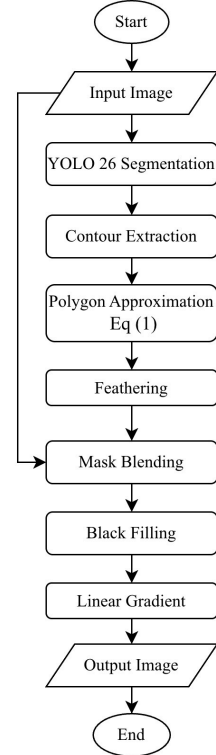


Fig. 2. Flowchart of the proposed method.

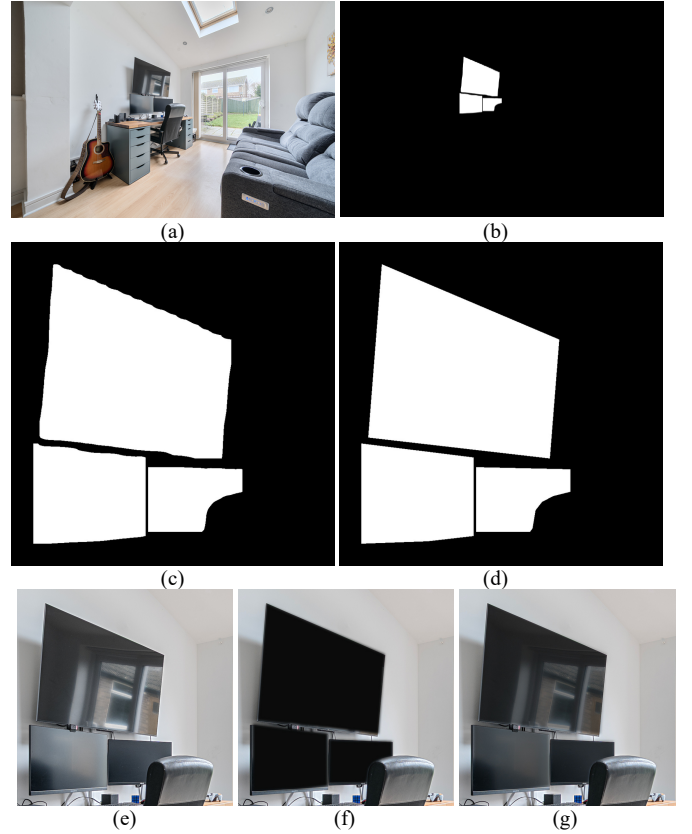


Fig. 3. Intermediate outputs of the proposed method: (a) original input image, (b) output of YOLO26 segmentation, (c) to (g) zoomed-in views around the TV screen area showing (c) YOLO26 segmentation, (d) result after polygon approximation, (e) original image, (f) result after black filling, and (g) final output after applying the linear gradient.

III. EXPERIMENTAL RESULTS

In this section, the performance of the proposed method is evaluated quantitatively and qualitatively. We first train the YOLO26 model for instance segmentation of TV screens. Of the annotated images, 70% are used for training, 20% for validation, and 10% for testing. Due to GPU memory limitations, the input image size is set to 1280×1280 pixels. In addition, the segmentation mask down sampling ratio is set to 4, which significantly accelerates training with only a negligible loss in accuracy. The YOLO26 model is initialized with weights pre-trained on the COCO dataset.

Based on a wide range of experiments, the hyperparameter settings that yielded the best performance are as follows: the Stochastic Gradient Descent (SGD) optimizer with a learning rate of 0.01, a momentum of 0.9, and a maximum number of detections of 10 per image. To enhance the model robustness against varying lighting conditions commonly encountered in real estate photographs, data augmentation is applied during training. Specifically, random brightness adjustments between -15% and +15% and random exposure adjustments between -10% and +10% are used. These augmentations help the model generalize better to different illumination scenarios without altering the geometric properties of the TV screens.

YOLO26 offers five model variants (nano, small, medium, large, and extra-large) with different trade-offs between accuracy and inference speed. Table I summarizes the segmentation performance of these variants on our TV screen dataset.

Table I. Segmentation performance of various models of YOLO26 on the test data.

Model	Precision	Recall	mAP50	mAP50-95	GFLOPs
n	0.864	0.839	0.883	0.837	9
s	0.919	0.868	0.928	0.860	34.1
m	0.906	0.895	0.947	0.871	121.2
l	0.937	0.895	0.942	0.864	139.4
x	0.971	0.894	0.956	0.871	313

In this Table, precision and recall are defined as follows:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

where TP , FP , and FN denote true positive, false positive, and false negative, respectively. The metric mAP refers to the mean average precision across all classes. Since the dataset contains only one class (TV screen), mAP is equivalent to AP (average precision). The AP metric is calculated as the area under the precision-recall curve at specific Intersection over Union (IoU) thresholds. Specifically, mAP50 corresponds to an IoU threshold of 0.5, while mAP50-95 represents the average AP across IoU thresholds ranging from 0.5 to 0.95 with a step size of 0.05. Additionally, GFLOPs stands for Giga Floating-point Operations and is used to measure the computational complexity of the model.

As shown in Table I, the extra-large model (YOLO26x) achieves the highest accuracy while the nano variant offers the fastest inference at the cost of lower accuracy. Given that high accuracy is critical for reliable TV screen blackout in real

estate images and considering that the extra-large model still maintains reasonable computational efficiency (313 GFLOPs), YOLO26x is selected as the base model for all subsequent experiments.

Fig. 4 shows the loss curves for training and validation on the dataset. Fig. 4 (a) depicts the segmentation loss, associated with the accuracy of predicted object boundaries (masks). Fig. 4 (b) shows the classification loss, a key component that drives the model to assign precise class probabilities to detected objects. Minimizing the classification loss enables the model to accurately label objects, thereby improving overall detection reliability. Overall, Fig. 4 demonstrates convergence of the model on the dataset. Early stabilization of validation losses suggests that the model successfully extracts generalizable features from the training set, enhancing its ability to make reliable predictions on new images.

Fig. 5 presents the precision, recall, mAP50, and mAP50-95 metrics on the validation set as functions of training epochs. These metrics provide a comprehensive assessment of the model detection and segmentation capabilities. All metrics exhibit a general upward trend with increasing epochs, indicating stable convergence of the YOLO26 model. Precision and recall both stabilize at high values after approximately 40 epochs, while mAP50 and mAP50-95 reach 0.955 and 0.870, respectively, by the end of training. These promising mAP values highlight the effectiveness and strong generalization capability of the proposed model for TV screen segmentation.

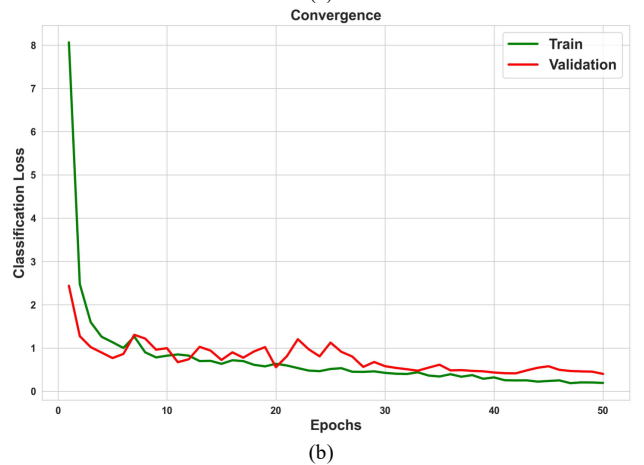
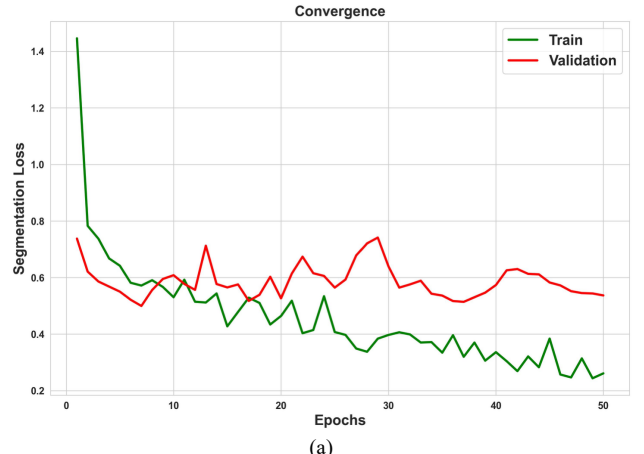


Fig. 4. Loss values as a function of epochs: (a) segmentation loss, and (b) classification loss.

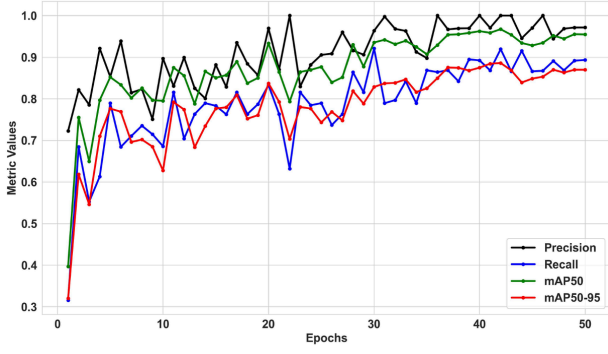


Fig.5. Precision, recall, mAP50, and mAP50-95 values on the validation set as function of epochs.

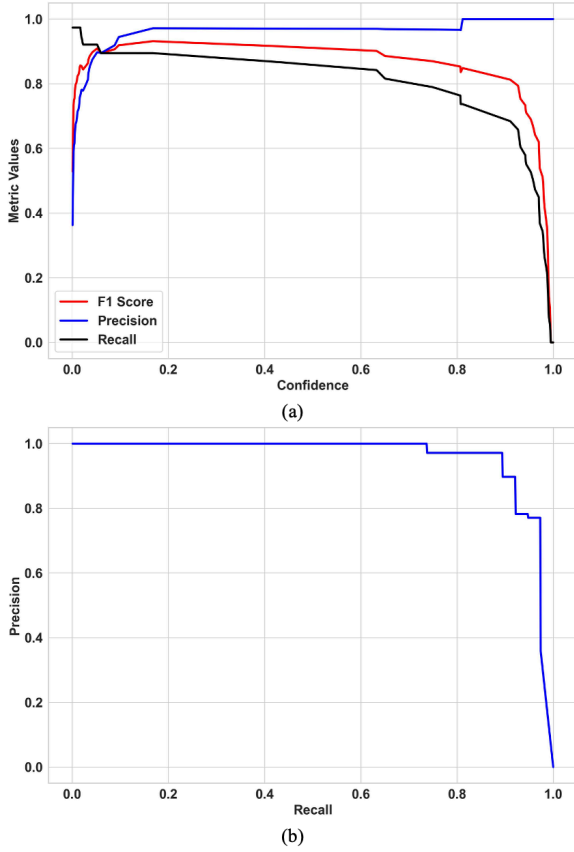


Fig. 6. Quantitative metrics of the YOLO26 segmentation model as a function of the confidence threshold.

Fig. 6 presents the quantitative performance of the proposed YOLO26 model across different confidence thresholds. Fig. 6 (a) shows the Precision, Recall, and F1-score curves as functions of the confidence threshold, where the F1-score is the harmonic mean of precision and recall. As observed, the model maintains high precision and F1-score

across a wide range of confidence values, while recall gradually decreases at higher thresholds. The high precision is particularly important in this work, as it indicates a low false positive rate. This is highly desirable given one of the main challenges in our dataset, the presence of dark rectangular objects that are not TV screens (see Fig. 1(c)), which can easily lead to false positives. Fig. 6 (b) illustrates the Precision-Recall curve, demonstrating a strong balance between precision and recall. The large area under this curve indicates robust segmentation performance, confirming the effectiveness of the trained model for TV screen detection in real estate images.

Fig. 7 illustrates the performance of the proposed pipeline on a challenging real estate image containing one TV screen along with several dark rectangular objects that can easily lead to false positives. As shown in Fig. 7(b), the trained YOLO26 model successfully detects the TV screens without any false positives. Fig. 7(c) and (d) present the initial YOLO26 segmentation output and the refined mask after polygon approximation, respectively. The zoomed-in views in Fig. 7(e)–(g) compare the original input image, the result edited by a professional manual editor, and the output of the proposed method. The visual comparison confirms that the blackout result produced by the proposed approach is very close to the professional manual editing, with negligible visible differences.

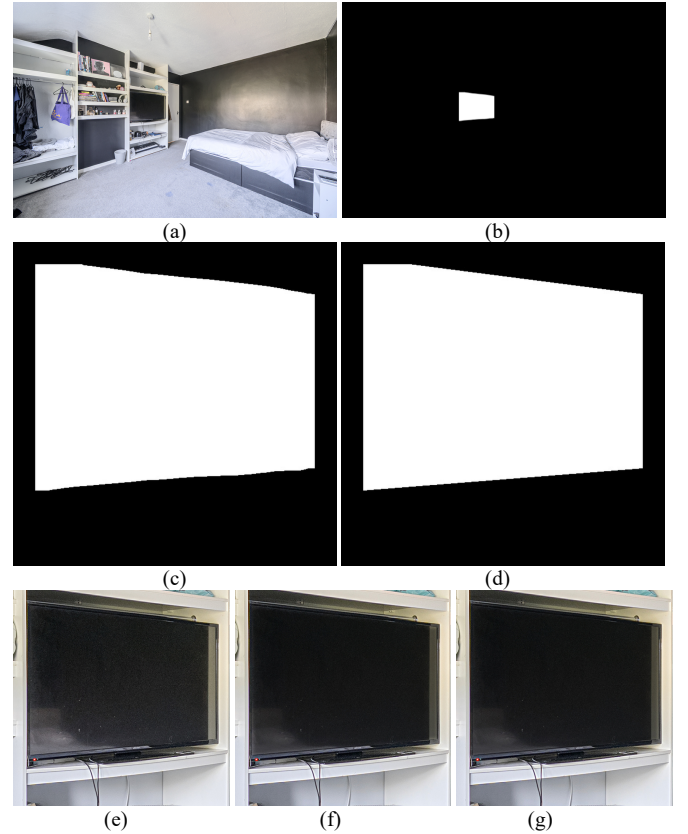


Fig. 7. Results of the proposed TV screen blackout pipeline: (a) original input image, (b) output of YOLO26 segmentation, (c) to (g) zoomed-in views around the TV screen area showing (c) YOLO26 segmentation, (d) result after polygon approximation, (e) original image, (f) TV blackout by a professional editor, and (g) TV blackout by the proposed method.

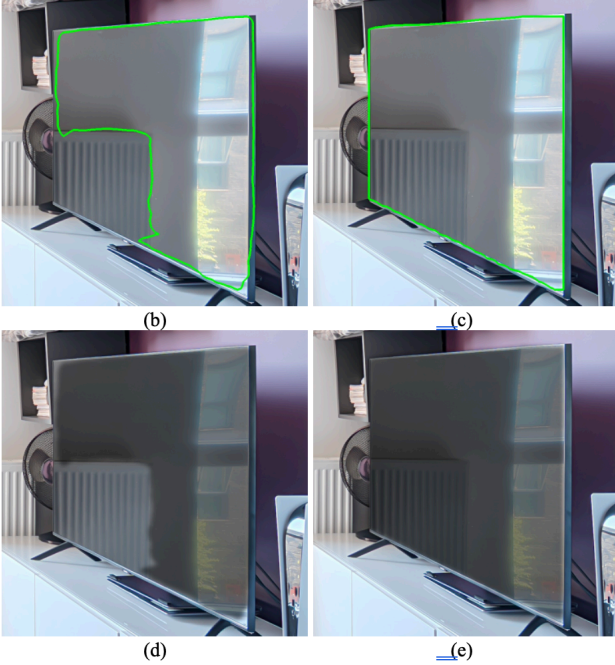
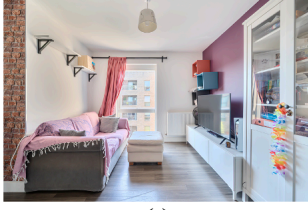


Fig. 8. TV screen blackout process: (a) original input image, (b)-(c) segmented TV screen boundaries using SAM3 and YOLO26, respectively, and (d)-(e) the corresponding final blackout results.

Fig. 8 compares the TV screen blackout results using two leading segmentation models, SAM3 [15] and YOLO26, on a typical real estate image. On our machine (Apple MacBook Pro with M4 Max chip), SAM3 takes 686 ms (milliseconds) while YOLO26 takes 143 ms, making YOLO26 approximately 4.8 times faster. The segmentation outputs are shown in Fig. 8 (b) and Fig. 8 (c). As can be observed, YOLO26 generates significantly more accurate boundaries than SAM3. The effect of segmentation quality is clearly visible in the final blackout results presented in Fig. 8 (d) and Fig. 8 (e). While SAM3 leaves portions of the TV screen unmasked (e.g., treating the reflected radiator as an occlusion), resulting in an unnatural and synthetic appearance, the YOLO26-based approach produces a clean and realistic blackout that closely resembles manual editing.

IV. CONCLUSION

An automatic TV screen blackout framework was proposed in this paper. It was based on training the YOLO26 model for instance segmentation using our comprehensive dataset of real estate images. After careful hyperparameter tuning, the model achieved near optimal performance. The

segmentation output was refined by polygon approximation of the contour, which preserved the predominant quadrilateral shape of the TV screen. In addition to black filling, a subtle linear gradient shadow was applied to create a more natural appearance that closely resembled professional manual editing. Experimental results demonstrated the effectiveness of the proposed approach, achieving a precision of 0.971, recall of 0.894, mAP50 of 0.956, and mAP50-95 of 0.871 on the validation set. Compared to SAM3, YOLO26 was approximately 4.8 times faster while producing more accurate boundaries and realistic blackout results.

REFERENCES

- [1] S. N. Bayrak, G. I. Alptekin, G. K. Orman, and A. Arslan, "Analyzing the impact of visual and listing features in real estate listings," in Proc. ICCSA Workshops, 2025.
- [2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2961–2969.
- [3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2016.
- [4] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring R-CNN," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 6409–6418.
- [5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. (MICCAI), 2015, pp. 234–241.
- [6] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv:1804.02767, 2018.
- [7] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv:2004.10934, 2020.
- [8] G. Jocher et al., "YOLOv5," Ultralytics, 2021. [Online]. Available: <https://github.com/ultralytics/yolov5>
- [9] G. Jocher, A. Chaurasia, and J. Qiu, "YOLOv8," Ultralytics, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [10] G. Jocher et al., "YOLOv10," Ultralytics, 2024. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [11] G. Jocher et al., "YOLOv11," Ultralytics, 2024. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [12] G. Jocher et al., "YOLO26," Ultralytics, 2026. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [13] A. Kirillov et al., "Segment anything," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2023, pp. 4015–4026.
- [14] N. Ravi et al., "SAM 2: Segment anything in images and videos," arXiv:2408.00714, 2024.
- [15] N. Carion et al., "SAM 3: Segment anything with concepts," arXiv:2511.16719, Nov. 2025.
- [16] B. Cheng et al., "Masked-attention mask transformer for universal image segmentation," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 1290–1299.
- [17] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2021, pp. 7262–7272.
- [18] N. Carion et al., "End-to-end object detection with transformers," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2020, pp. 213–229.
- [19] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," arXiv:2010.04159, 2020.
- [20] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature," The Canadian Cartographer, vol. 10, no. 2, pp. 112–122, 1973.