Virtual Reality
https://doi.org/10.1007/s10055-023-00858-0

ORIGINAL ARTICLE

Lightweight real-time hand segmentation leveraging MediaPipe landmark detection

Guillermo Sánchez-Brizuela[1] · Ana Cisnal[1] · Eusebio de la Fuente-López[1] · Juan-Carlos Fraile[1] · Javier Pérez-Turiel[1]

Received: 19 December 2022 / Accepted: 21 August 2023

Abstract

Real-time hand segmentation is a key process in applications that require human–computer interaction, such as gesture recognition or augmented reality systems. However, the infinite shapes and orientations that hands can adopt, their variability in skin pigmentation and the self-occlusions that continuously appear in images make hand segmentation a truly complex problem, especially with uncontrolled lighting conditions and backgrounds. The development of robust, real-time hand segmentation algorithms is essential to achieve immersive augmented reality and mixed reality experiences by correctly interpreting collisions and occlusions. In this paper, we present a simple but powerful algorithm based on the MediaPipe Hands solution, a highly optimized neural network. The algorithm processes the landmarks provided by MediaPipe using morphological and logical operators to obtain the masks that allow dynamic updating of the skin color model. Different experiments were carried out comparing the influence of the color space on skin segmentation, with the CIELab color space chosen as the best option. An average intersection over union of 0.869 was achieved on the demanding Ego2Hands dataset running at 90 frames per second on a conventional computer without any hardware acceleration. Finally, the proposed segmentation procedure was implemented in an augmented reality application to add hand occlusion for improved user immersion. An open-source implementation of the algorithm is publicly available at https://github.com/itap-robotica-medica/lightweight-hand-segmentation.

Keywords Augmented reality · Hand segmentation · MediaPipe · Online processing · Semantic segmentation

1 Introduction

Hand usage is a fundamental element of human interaction with the world. Consequently, hands play a crucial role in mixed reality applications that try to mimic real-life experiences. The user's immersion in these applications is highly conditioned on the accurate detection and representation of the hands, as hands are the body part most often interacting with the virtual elements.

Researchers have studied and proposed different methods to locate and characterize hands in multimedia content such as videos or images. The approaches employed range from traditional computer vision techniques to convolution-based deep learning models and can be divided into two main groups, hand tracking and hand segmentation methods.

Hand tracking methods deal with locating key features of the hand to provide a set of identifiable landmarks that are used to model the position of the palm and fingers. Common approaches to this problem include optimized convolutional neural networks to track hands in images (Zhang et al. 2020; Lim et al. 2020) or employing specialized hardware such as dedicated sensors (Glauser et al. 2019) and depth-sensing cameras (Chakraborty et al. 2018).

Conversely, hand segmentation methods consist of extracting the pixels belonging to the hands that appear in the image. A plethora of methods for skin color detection

✉ Ana Cisnal
ana.cisnal@uva.es

Guillermo Sánchez-Brizuela
guillermo.sanchez.brizuela@uva.es

Eusebio de la Fuente-López
efuente@uva.es

Juan-Carlos Fraile
jcfraile@eii.uva.es

Javier Pérez-Turiel
jpturiel@uva.es

[1] ITAP (Instituto de las Tecnologías Avanzadas de la Producción), Universidad de Valladolid, Paseo del Cauce 59, 47011 Valladolid, Spain

Published online: 05 September 2023

exist in the literature. The most popular technique is color thresholding using different color spaces, especially RGB, HSV and YCbCr. Color thresholding methods allow fast processing; however, these purely color-based techniques have limitations. The wide variety of existing skin pigmentations, changes in illumination that significantly modify the initial colors or the presence of skin-colored objects that can be detected as hands are some of their major drawbacks (Kang et al. 2017). Although improvements have been made in more recent approaches by creating and dynamically updating the skin appearance model in different color spaces (Zhang et al. 2018; Zhao et al. 2018; Thwe and Yu 2019), inaccuracies still appear in the results.

Another practice is the extraction of information at both the pixel and superpixel level for use as input to a tree-based machine learning classifier (Zhao and Quan 2018; Zhu et al. 2015; Baraldi et al. 2015). Superpixels are obtained by over segmenting the image with algorithms such as the simple linear iterative clustering SLIC algorithm (Achanta et al. 2012). Finally, more recent approaches include the use of convolutional deep learning architectures specifically designed for segmentation (which have already been proven successful in widely varied tasks (Yu et al. 2021, 2022; Liu et al. 2020)) such as UNet (Tsai and Huang 2022; Wang et al. 2019), OR-Skip-Net (Arsalan et al. 2020) or networks using Bayesian techniques (Cai et al. 2020).

In this paper, we propose a novel algorithm for segmenting hands in video frames that is based on the information provided by a highly optimized neural network called MediaPipe Hands. MediaPipe Hands is a palm and finger tracking solution introduced by Zhang et al. (2020). This solution consists of a highly optimized pipeline composed of two models, a hand palm detector that provides an oriented bounding box of the hand and a hand landmark model that operates on the bounding box to obtain 2.5D landmarks. These landmarks are composed of the x- and y-coordinates in the image space of 21 key points of the hand, along with an additional z-coordinate; the depth of each landmark is relative to the wrist key point (Fig. 1b).

This hand tracking solution has been successfully used in diverse tasks, such as sign language recognition (Cheng et al. 2020; Shin et al. 2021), hand 3D model reconstruction (Seeber et al. 2021) and gesture-based control in rehabilitation systems leveraging the 2.5D hand pose (Xiao et al. 2023). However, the positional landmarks provided by MediaPipe Hands are not suitable for augmented or mixed reality applications, as in these cases, pixel-level hand segmentation is required to cope with hand collisions and occlusions.

As noted, current segmentation approaches that work with monocular cameras use either superpixel classifiers or convolutional neural networks, and both require specialized hardware and nontrivial optimizations to operate in real time. In this paper, we propose extracting the hands in video frames in real time by processing the landmarks provided by MediaPipe with morphological and logical operators (Fig. 1). Our solution, although conceptually simple, allows the implementation of hand segmentation in mixed reality applications running on nonaccelerated mobile and monocular hardware.

2 Proposed algorithm

The algorithm we present in this paper is a pipeline of six stages (Fig. 2) based on a calculation of dynamic color ranges to model skin color in a given color space.

1. The first stage is the extraction of characteristic hand landmarks using MediaPipe Hands. The extracted landmarks are used to draw a skeleton of the hand.
2. The skeleton is binarized to generate a skeleton mask (M1). Another mask (Mask M2) is generated through a strong dilation of this first mask, iterating twice with a square kernel (k) of size 21×21 pixels as defined in Eq. 1. This second mask serves the function of limiting the area of the image where the hand boundaries may be present. To choose the size for this dilation kernel, we carried out a grid search on the training set of the dataset and selected the best-performing parameters.

$$M2 = (M1 \oplus k) \oplus k \tag{1}$$

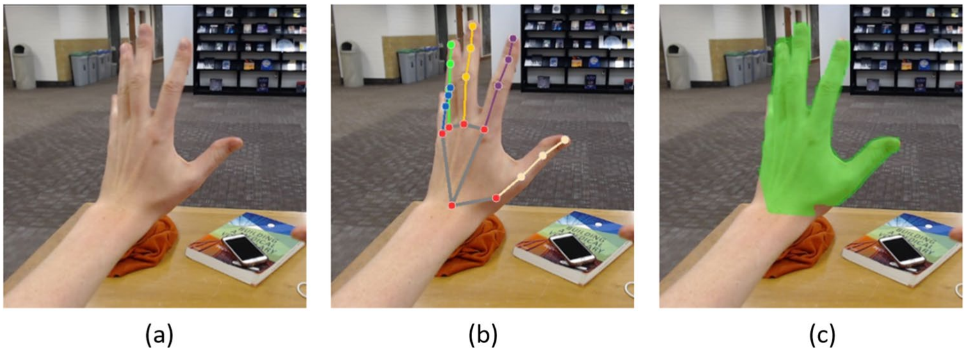

Fig. 1 Algorithm stages. **a** Input image. **b** MediaPipe hands output. **c** Proposed segmentation solution results

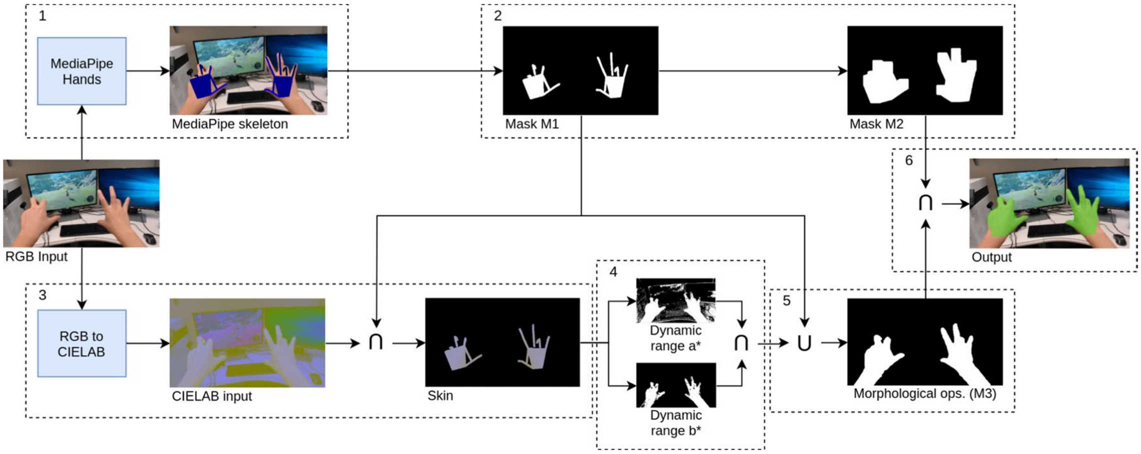

Fig. 2 Segmentation pipeline stages

3. The input image is converted to the CIELab color space and masked with M1 (Eq. 2). This color space provides a more suitable representation of skin-like colors for segmentation compared to other color spaces (Ly et al. 2020).

$$\text{Skin} = \text{RGB2Lab}(\text{Input}) \cap M1 \tag{2}$$

4. The result from the previous masking is used to randomly sample 1500 values that correspond to the hands in both the a* and b* channels. These sampled values are later sorted to obtain the first and third quartiles. These percentiles act as the boundaries to a region in the CIELab color space where the majority of the pixels corresponding to hands in the same frame are expected to be located. Once these boundaries have been calculated, two more channelwise masks (Ma and Mb) are calculated on the a* and b* channels as presented in Eqs. 3 and 4, eliminating every value that falls out of the calculated ranges and keeping the values of the pixels (p) that fall within those ranges. Another binary mask Mab is calculated as the intersection of Ma and Mb (Eq. 5), ensuring that only the pixels in both ranges are selected.

$$Ma = \{p \in (\text{Skin}_{a*(Q1)}, \text{Skin}_{a*(Q3)})\} \tag{3}$$

$$Mb = \{p \in (\text{Skin}_{b*(Q1)}, \text{Skin}_{b*(Q3)})\} \tag{4}$$

$$\text{Mab} = Ma \cap Mb \tag{5}$$

5. After this union, a morphological operation of closing followed by an opening is performed on the union of Mab and M1 to fill holes, reduce noise and smooth the resulting mask. This resulting mask is represented in Fig. 2 as M3. Given a square kernel k, this operation can be expressed as Eq. 6.

$$M3 = ((((\text{Mab} \cup M1) \oplus k) \ominus k) \ominus k) \oplus k \tag{6}$$

6. Finally, to avoid background false positives, the M2 mask is used to eliminate any blob that is detected outside of the hand area. Consequently, the result of the segmentation is given by the mask bitwise expression outlined in Eq. 7.

$$\text{Output} = M3 \cap M2 \tag{7}$$

3 Dataset and evaluation

Multiple datasets have focused on the task of hand segmentation (Bambach et al. 2015; Shilkrot et al. 2019; Khan and Borji 2018). In this work, we use Ego2Hands (Lin and Martinez 2020) as the study set of images. This dataset is composed of frames from egocentric videos of different hands in movement and their corresponding segmentation annotations (Fig. 3).

These videos have been recorded in different environments with distinct illumination and background conditions. Furthermore, different skin tones are also present. The training set of this dataset has been established to design and parametrize the algorithm. The evaluation set has been isolated and has only been used to evaluate the metrics shown in this document.

To formally evaluate the performance of our algorithm, we calculate the intersection over union (IoU) of the predicted mask and the annotation. This metric was selected due to its widespread acceptance as a standard measure in the field of image segmentation and its ease of use when comparing our results with existing and future literature. In this evaluation, we masked the annotation with the M2

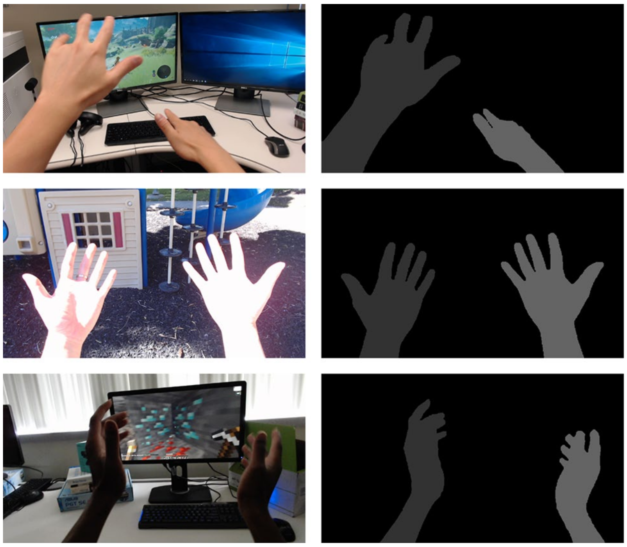

Fig. 3 Random samples from the evaluation sequences of Ego2Hands (Lin and Martinez 2020)

mask. This last operation fulfills the dual role of removing the forearm region from the annotation and conditioning the measure to frames where MediaPipe detects something. However, we make no distinction based on the correctness of the MediaPipe landmarks if they are detected.

Additionally, we present a performance and execution-time study of the algorithm when executed on an Intel® i5-11600K @ 3.90 GHz CPU, with no hardware acceleration or GPU utilization.

4 Results

In the following section, we include the quantitative and qualitative results obtained following the previously described evaluation methodology, including a study of the influence of the color space choice to model hand appearance and a measure of the computation time each stage of the algorithm takes.

4.1 Color space influence

During the development of this research, we carried out tests with different parametrizations and configurations of the algorithm, more specifically, on the space color from which the random samples are taken. For dynamic range filtering, we explore the impact of choosing the CIELab, HSV or YCrCb color spaces using intersection over union (IoU). In these experiments, the chosen percentiles are (40, 60), (2, 98) and (2, 98) for the hue, saturation and value channels and (1, 99), (35, 65) and (35, 65) for the Y, Cb and Cr channels, respectively.

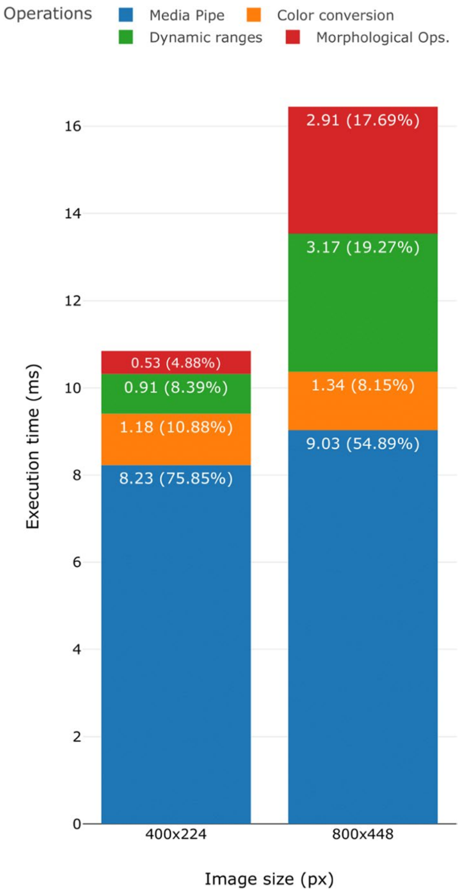

Fig. 4 Execution-time decomposition based on input size

The results for the eight sequences in the evaluation set of the Ego2Hands dataset are presented in Table 1, where it can be observed that the CIELab color space yields better results in most of the video sequences, closely followed by the YCrCb color space. HSV, however, falls behind with slightly but clearly worse results (except for sequence 5, characterized by a notably dark ambient illumination).

4.2 Performance study

A decomposition of the algorithm operations with their compute times averaged over 250 images of size 800×448 and 400×224 is presented in Fig. 4. These image resolutions were chosen as the first represents the original size of the images in the dataset and the second, where we reduced the image size to 50% in each spatial dimension, provided a sensible balance between accuracy loss and processing speed gains. The mean execution times of a single image are 16.45 ms and 10.85 ms, respectively.

With images of size 800×448, MediaPipe represents the majority (9.03 ms, 54.89% of the total) of the processing time, while the operations added for segmentations are

Table 1 Comparison of IoU in Ego2Hands evaluation sequences sampling in different color spaces

	Seq1	Seq2	Seq3	Seq4	Seq5	Seq6	Seq7	Seq8	Mean
CIELab	**0.915**	0.833	**0.889**	**0.898**	0.758	**0.886**	**0.876**	**0.842**	**0.869**
YCrCb	0.906	**0.840**	**0.889**	0.896	0.760	0.876	0.875	0.841	0.866
HSV	0.844	0.838	0.868	0.853	**0.837**	0.859	0.866	0.817	0.849

The values in bold indicate the highest IoU achieved in each sequence among the three color spaces considered during the evaluation

carried out in 7.42 ms. Translating these measures to FPS, the MediaPipe processing alone can be executed at approximately 110 frames per second; however, the overhead of segmentation reduces this metric to the value of 60.79 frames per second.

Furthermore, if the input images are reduced to a size of 400×224 pixels, which still perfectly encompasses the hand localization information, the segmentation overhead diminishes drastically, as shown in Fig. 4. With the reduced images, while the processing time associated with MediaPipe and the color conversions does not decrease substantially, the execution time of the dynamic ranges and the morphological operations is greatly reduced from 6.08 to 1.44 ms. The resulting processing time is 10.85 ms, increasing the rate at which the CPU can perform segmentation to slightly more than 90 FPS (for reference, the MediaPipe Landmark detection, processes the images at a rate of 121.5 FPS).

4.3 Qualitative results

Figure 5 shows the mask produced by the algorithm below the annotation masked with the M2 mask, as described in Sect. 3. In the results below, several characteristics of our proposal can be appreciated. Overall, the different skin tones and lighting conditions demonstrate the robustness added by the dynamically calculated ranges. Additionally, Fig. 5a, d shows the algorithm's ability to separate hand borders and similar color backgrounds in nonextreme cases, and finally, Fig. 5b and Fig. 5c illustrates that the algorithm can also deal with blurriness caused by hand movements.

5 Discussion

The performance evaluation of the algorithm using different color spaces revealed that CIELab achieved the best performance on the evaluation set (IoU = 0.869). While the YCrCb performed similarly, it was less accurate (IoU = 0.866), and the HSV performed worst (IoU = 0.849). These results are consistent with previous research in the field.

Montenegro et al. (2013) conducted a comparative study of color spaces for detecting human skin using a probabilistic classifier. They evaluated the quality of the results using the Matthews correlation coefficient (MCC) and found that CIELab performed best (MCC = 0.074) with the YCbCr performing similarly but less stably (MCC = 0.779), and

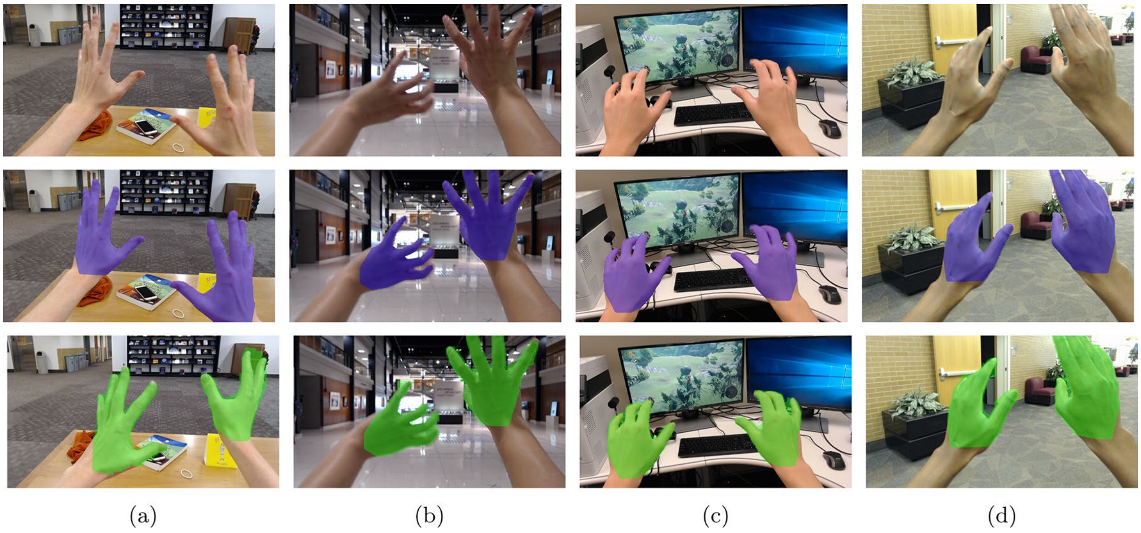

Fig. 5 Qualitative results of the proposed algorithm. (Top): Input image. (Center): Intersection of the ground truth annotation and the M2 mask. (Bottom): Proposed segmentation solution result

HSV yielding the worst performance (MCC = 0.736). Similarly, Kaur and Kranthi (2012) concluded that skin color segmentation with CIELab color space was better than YCbCr. Likewise, Li and Kitani (2013) performed experiments for hand detection using RGB, HSV and CIELab color spaces, with the latter achieving the best performance.

The CIELab color space is based on the opponent color model of human vision and is device independent. Moreover, the fact that Lightness is a separate channel in CIELab makes it easier to identify the chromaticity of skin tones, which usually lie within a relatively narrow range of values in the A and B channels, reducing its sensitivity to errors. These attributes could make the CIELab color space the most adequate for detecting human skin using the proposed algorithm.

On the other hand, a fast and robust hand segmentation is the first challenge and a necessary step in many real-time implementations in which hand-object interaction is required, such as augmented reality, medial application and human–robot interaction. The main advantage of our algorithm when compared to deep learning or superpixel classification techniques is its ability to be executed in real time without dedicated hardware or special optimizations.

While it is true that the segmentation overhead almost doubles the processing time of MediaPipe with images of size 800x448, considering the highly optimized nature of the hand landmark detection platform, this is still remarkably fast when considering the resulting quality, the platform running the algorithm (CPU) and the ease of implementation of the algorithm. Therefore, we consider that the proposed algorithm achieves a substantially positive trade-off between segmentation capabilities and computational cost, particularly when compared with the current traditional computer vision approaches with similar results that take up to 3497 milliseconds (Zhao et al. 2018) when running on an Intel i7-4500U (mainly due to the computation of superpixels to dynamically adjust the thresholding).

Additionally, in order to demonstrate and test the real-time performance of the developed segmentation algorithm, an augmented reality application that includes hand occlusion has been implemented. In this application, a Unity3D scene interacts with the Python implementation of the proposed algorithm. This Python service processes real-time input from a camera, segments the hand regions and sends both images (frame and mask) to the Unity3D application, where the real images and virtual elements are fused in a final frame (Fig. 6). This augmented reality environment illustrates that the developed hand segmentation algorithm can improve, at the current stage, the immersion and the quality of interactive capabilities in mixed reality applications.

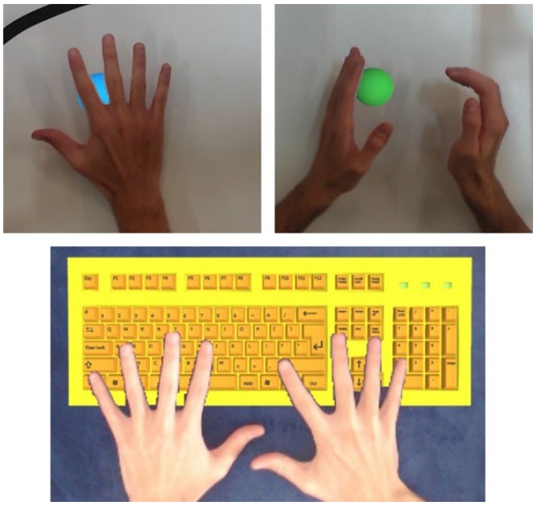

Fig. 6 Real-time screenshots of hand occlusion in an augmented reality application using our algorithm. The color spheres and yellow keyboards are virtual (colour figure online)

Finally, there are two main weaknesses that may affect the quality of segmentation due to the nature of the algorithm. First, the dependence on MediaPipe, whose performance directly impacts the sampling operation. In a worst-case scenario where MediaPipe does not detect any hand (Fig. 7a) and therefore does not supply any landmarks to the second stage of the algorithm, the segmentation result will be null. On the other hand, if MediaPipe produces deficient landmarks (Fig. 7b), the selected color ranges could also be affected depending on the magnitude of error of the landmarks. This last situation, however, is mitigated by the percentile-based channelwise ranges. Second, while the dynamic color range selection partially solves it, the algorithm still works using color spaces and, consequently, can malfunction in situations with extreme lighting conditions such as strong shadows or sensor saturation (Fig. 7c) and backgrounds of similar colors to those sampled in the hands (Fig. 7d). Again, this is mitigated in our algorithm with the M2 mask, limiting the segmentation result to the area surrounding each hand.

6 Conclusions

In this paper, we propose a hand segmentation algorithm built on top of the highly optimized MediaPipe Hand Localization platform. We study different color spaces and attain a mean IoU of 0.869 on the demanding Ego2Hands dataset evaluation set using the CIELab color space. Additionally, we demonstrate that our solution can be easily implemented in Python to run at more than 90 frames per second without hardware acceleration or highly optimized code. Finally, we prove the value of the algorithm by using it to add hand

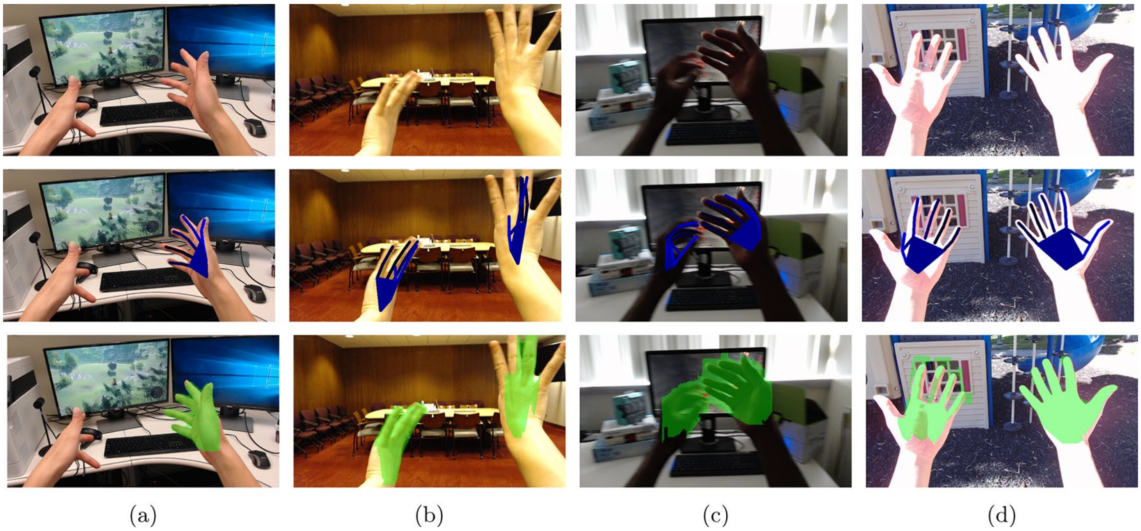

Fig. 7 Segmentation algorithm failure cases. (Top): Input image. (Center): MediaPipe Hands output. (Bottom): Proposed segmentation solution result

occlusion to an augmented reality environment, increasing the immersion of the user.

While we consider the results to present a very good trade-off between quality and execution time, further research could explore the utilization of spatial and temporal information in video processing to increase the result accuracy and to add robustness to the background conditions. Furthermore, optimization strategies and mobile or embedded hardware implementations could also be developed based on this work.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature. The research leading to these results received funding from the Ministry of Science and Innovation of Spain under Grant Agreement No. RTC2019-007350-1.

Data availability Every image used in this work to find the optimum parametrization of the algorithm, as well as to evaluate the resulting models, is part of the Ego2Hands dataset, which is publicly available at https://github.com/AlextheEngineer/Ego2Hands.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Süsstrunk S (2012) Slic superpixels compared to state-of-the-art superpixel methods. IEEE Trans Pattern Anal Mach Intell 34(11):2274–2282. https://doi.org/10.1109/TPAMI.2012.120

Arsalan M, Kim DS, Owais M, Park KR (2020) Or-skip-net: outer residual skip network for skin segmentation in non-ideal situations. Expert Syst Appl 141:112922. https://doi.org/10.1016/j.eswa.2019.112922

Bambach S, Lee S, Crandall DJ, Yu C (2015) Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In: 2015 IEEE International conference on computer vision (ICCV), pp. 1949–1957. https://doi.org/10.1109/ICCV.2015.226

Baraldi L, Paci F, Serra G, Benini L, Cucchiara R (2015) Gesture recognition using wearable vision sensors to enhance visitors' museum experiences. IEEE Sens J 15(5):2705–2714. https://doi.org/10.1109/JSEN.2015.2411994

Cai M, Lu F, Sato Y (2020) Generalizing hand segmentation in egocentric videos with uncertainty-guided model adaptation. In: 2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp. 14380–14389. https://doi.org/10.1109/CVPR42600.2020.01440

Chakraborty BK, Sarma D, Bhuyan MK, MacDorman KF (2018) Review of constraints on vision-based gesture recognition for human-computer interaction. IET Comput Vis 12(1):3–15. https://doi.org/10.1049/iet-cvi.2017.0052

Cheng J, Wei F, Liu Y, Li C, Chen Q, Chen X (2020) Chinese sign language recognition based on dtw-distance-mapping features. Math Probl Eng 2020:1–13. https://doi.org/10.1155/2020/8953670

Glauser O, Wu S, Panozzo D, Hilliges O, Sorkine-Hornung O (2019) Interactive hand pose estimation using a stretch-sensing soft glove. ACM Trans Graph 10(1145/3306346):3322957

Kang B, Tan K-H, Jiang N, Tai H-S, Tretter D, Nguyen T (2017) Hand segmentation for hand-object interaction from depth map. In: 2017 IEEE global conference on signal and information processing (GlobalSIP), pp. 259–263. https://doi.org/10.1109/GlobalSIP.2017.8308644

Kaur A, Kranthi B (2012) Comparison between ycbcr color space and cielab color space for skin color segmentation. Int J Appl Inf Syst 3(4):30–33

Khan AU, Borji A (2018) Analysis of hand segmentation in the wild. In: 2018 IEEE/CVF conference on computer vision and pattern recognition, pp. 4710–4719. https://doi.org/10.1109/CVPR.2018.00495

Li C, Kitani KM (2013) Pixel-level hand detection in ego-centric videos. In: 2013 IEEE conference on computer vision and pattern recognition, pp. 3570–3577. https://doi.org/10.1109/CVPR.2013.458

Lim G, Jatesiktat P, Ang W (2020) MobileHand: real-time 3D hand shape and pose estimation from color image, pp. 450–459. https://doi.org/10.1007/978-3-030-63820-7_52

Lin F, Martinez TR (2020) Ego2hands: a dataset for egocentric two-hand segmentation and detection. ArXiv arXiv:2011.07252

Liu X, Zhu X, Li M, Wang L, Zhu E, Liu T, Kloft M, Shen D, Yin J, Gao W (2020) Multiple kernel *k*k-means with incomplete kernels. IEEE Trans Pattern Anal Mach Intell 42(5):1191–1204. https://doi.org/10.1109/TPAMI.2019.2892416

Ly BCK, Dyer EB, Feig JL, Chien AL, Del Bino S (2020) Research techniques made simple: cutaneous colorimetry: a reliable technique for objective skin color measurement. J Investig Dermatol 140(1):3–121. https://doi.org/10.1016/j.jid.2019.11.003

Montenegro J, Gómez W, Sánchez-Orellana P (2013) A comparative study of color spaces in skin-based face segmentation. In: 2013 10th International conference on electrical engineering, computing science and automatic control (CCE), pp. 313–317. https://doi.org/10.1109/ICEEE.2013.6676048

Seeber M, Oswald MR, Poranne R (2021) Realistichands: a hybrid model for 3d hand reconstruction. In: 2021 International conference on 3D vision (3DV). 22–31

Shilkrot R, Narasimhaswamy S, Vazir S, Nguyen MH (2019) Working hands: a hand-tool assembly dataset for image segmentation and activity mining. In: Proceedings of the British machine vision conference (BMVC), pp. 1–12. https://doi.org/10.5244/C.33.171

Shin J, Matsuoka A, Hasan MAM, Srizon AY (2021) American sign language alphabet recognition by extracting feature from hand pose estimation. Sensors. https://doi.org/10.3390/s21175856

Thwe PM, Yu MT (2019) Analysis on skin colour model using adaptive threshold values for hand segmentation. Int J Image Graph Signal Process 11(9):25–33. https://doi.org/10.5815/ijigsp.2019.09.03

Tsai T-H, Huang S-A (2022) Refined u-net: a new semantic technique on hand segmentation. Neurocomputing 495:1–10. https://doi.org/10.1016/j.neucom.2022.04.079

Wang W, Yu K, Hugonot J, Fua P, Salzmann M (2019) Recurrent u-net for resource-constrained segmentation. In: 2019 IEEE/CVF International conference on computer vision (ICCV), pp. 2142–2151. IEEE Computer Society, Los Alamitos, CA, USA. https://doi.org/10.1109/ICCV.2019.00223

Xiao F, Zhang Z, Liu C, Wang Y (2023) Human motion intention recognition method with visual, audio, and surface electromyography modalities for a mechanical hand in different environments. Biomed Signal Process Control 79:104089. https://doi.org/10.1016/j.bspc.2022.104089

Yu X, Lu Y, Gao Q (2021) Pipeline image diagnosis algorithm based on neural immune ensemble learning. Int J Press Vessels Pip 189:104249. https://doi.org/10.1016/j.ijpvp.2020.104249

Yu X, Ye X, Zhang S (2022) Floating pollutant image target extraction algorithm based on immune extremum region. Digit Signal Process. https://doi.org/10.1016/j.dsp.2022.103442

Zhang Q, Yang M, Kpalma K, Zheng Q, Zhang X (2018) Segmentation of hand posture against complex backgrounds based on saliency and skin colour detection. IAENG Int J Comput Sci 45(3):435–444

Zhang F, Bazarevsky V, Vakunov A, Tkachenka A, Sung G, Chang C-L, Grundmann M (2020) Mediapipe hands: on-device real-time hand tracking. arXiv arXiv:2006.10214

Zhao YL, Quan C (2018) Coarse-to-fine online learning for hand segmentation in egocentric video. J Image Video Proc. https://doi.org/10.1186/s13640-018-0262-1

Zhao Y, Luo Z, Quan C (2018) Coarse-to-fine online learning for hand segmentation in egocentric video. EURASIP J Image Video Process 2018:20. https://doi.org/10.1186/s13640-018-0262-1

Zhu X, Jia X, Wong K-YK (2015) Structured forests for pixel-level hand detection and hand part labelling. Comput Vis Image Underst 141:95–107. https://doi.org/10.1016/j.cviu.2015.07.008

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.