Implementation of A Skin Segmentation Model Based on Synthetic Face Images Using Deep Learning

Master thesis in the department of Electrical Engineering and Information Technology
by Kunzhi Shi (Student ID: 2898029)
Date of submission: June 4, 2025

1. Review: Prof. Dr.-Ing. Stefan Göbel, Serious Games
2. Review: Prof. Dr.-Ing. Christoph Hoog Antink, KIS*MED
3. Review: Maurice Rohr, M.Sc., KIS*MED
Darmstadt

TECHNISCHE UNIVERSITÄT DARMSTADT

Electrical Engineering and Information Technology Department

KIS*MED - AI Systems in Medicine

Studiengang Autonome Systeme

Erklärung zur Abschlussarbeit gemäß § 22 Abs. 7 APB TU Darmstadt

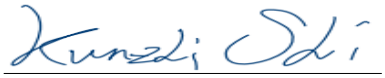
Hiermit erkläre ich, Kunzhi Shi, dass ich die vorliegende Arbeit gemäß § 22 Abs. 7 APB der TU Darmstadt selbstständig, ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt habe. Ich habe mit Ausnahme der zitierten Literatur und anderer in der Arbeit genannter Quellen keine fremden Hilfsmittel benutzt. Die von mir bei der Anfertigung dieser wissenschaftlichen Arbeit wörtlich oder inhaltlich benutzte Literatur und alle anderen Quellen habe ich im Text deutlich gekennzeichnet und gesondert aufgeführt. Dies gilt auch für Quellen oder Hilfsmittel aus dem Internet.

Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§ 38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

Darmstadt, 4. Juni 2025



Kunzhi Shi

Abstract

Training skin segmentation models using synthetic face images has garnered increasing attention as a promising alternative to real data, addressing challenges such as data scarcity, privacy concerns, and annotation costs. Skin segmentation plays a critical role in various computer vision applications, including facial recognition, medical imaging, and augmented reality. In this work, we implement and compare the performance of different segmentation network architectures trained solely on synthetic face images. The primary objective is to explore the feasibility of using synthetic data as a reliable substitute for real-world data in skin segmentation tasks. The study investigates a U-Net architecture with an EfficientNetB0 backbone and evaluates different training strategies based on varying numbers of segmentation annotations to assess their influence on model performance. Additionally, several state-of-the-art alternative architectures are examined to determine the most effective approach for skin segmentation. The model is validated on real-world images and subjected to stress testing to evaluate its robustness in diverse conditions. Furthermore, the best-performing model is integrated into a video segmentation pipeline and assessed for its effectiveness in Photoplethysmographic Imaging (PPGI) applications using multiple public and commonly used datasets, achieving performance that closely approaches that of models trained on real facial images. Experimental results demonstrate the potential of synthetic data in training robust skin segmentation models, providing valuable insights into its applicability for real-world scenarios. In order to ensure reproducibility and accessibility, the source code has been uploaded to the website: https://github.com/Supcode123/Skin-Segmentation-4-iPPG.

Contents

1. Introduction

Skin segmentation is a critical task in computer vision, with applications spanning facial recognition, medical imaging, and augmented reality. Accurate detection of skin regions is essential for extracting physiological signals, particularly in tasks such as Photoplethysmographic Imaging (PPGI), where the signal quality depends heavily on precise skin segmentation [1]. However, training effective segmentation models typically requires large annotated datasets, which are often difficult to obtain, especially due to privacy concerns and the high costs associated with annotation. Recent advances in deep learning have highlighted the potential of synthetic data for training robust models, offering a promising alternative to real-world data. Using synthetic face images can address data scarcity, reduce privacy concerns, and eliminate the need for labor-intensive annotation [2]. This work explores the feasibility of using synthetic face images for training skin segmentation models and evaluates the performance of various deep learning architectures in this context. One of the key challenges when using synthetic data is ensuring that the models generalize well to real-world scenarios. This research addresses this challenge by implementing and comparing different deep learning models trained on synthetic data, exploring how varying the number of segmentation labels impacts model performance, and assessing their ability to generalize to real-world images. By exploring these factors, we aim to contribute to the development of more robust and effective skin segmentation models that can be applied to real-world use cases, especially in fields like PPGI. The results of this study have the potential to drive further advancements in both synthetic data generation and the application of deep learning to skin segmentation tasks, making it a valuable contribution to the computer vision community.

1.1 Motivation

PPGI is a non-contact method used to extract blood volume pulse (BVP) signals from facial video, enabling non-invasive heart rate estimation. It has significant healthcare applications

such as remote heart rate monitoring, stress detection, and cardiovascular assessment. BVP extraction relies on analyzing small intensity changes in light absorption in the upper skin layers. However, dynamic effects like shadows and head movements can interfere, making precise skin segmentation crucial for accurate PPGI-based physiological signal extraction [3]. Traditional skin segmentation models are trained on real-world datasets, but these datasets suffer from several limitations, including data scarcity, privacy concerns, and high annotation costs. To address these challenges, synthetic data—generated through computer-rendered face models—has been proposed as a scalable and annotation-free alternative. However, whether models trained solely on synthetic data can generalize well to real-world PPGI applications remains an open question. This study aims to investigate the feasibility of using synthetic face images to train robust skin segmentation models for PPGI tasks. Specifically, we evaluate different segmentation network architectures, explore various training strategies involving different numbers of segmentation masks, and assess the models on real-world images and video-based physiological monitoring tasks. If synthetic data proves to be an effective substitute, it could significantly improve the accessibility and scalability of training segmentation models, ultimately enhancing the reliability of contactless physiological monitoring in medical applications.

1.2 Problem Statement and Contribution

Despite the potential of synthetic data, its use in skin segmentation for PPGI tasks remains underexplored, particularly in assessing how well models trained on synthetic data generalize to real-world images. The primary problem addressed in this work is the lack of comprehensive studies on the effectiveness of synthetic data in training skin segmentation models for PPGI. Additionally, there is a need to evaluate various training strategies, including the use of different numbers of segmentation masks, to determine their impact on model performance in extracting BVP signals and computing heart rate.

The key contributions of this work include:

- The implementation and comparison of multiple state-of-the-art segmentation network architectures trained exclusively on synthetic face images.
- An exploration of different training strategies based on varying numbers of labels in segmentation mask to assess their influence on model performance.
- A comprehensive evaluation of the trained models on real-world images to assess their generalization ability.

- Integration of the best-performing model into a video segmentation pipeline for PPGI applications, where BVP signals of patients are extracted and heart rate is computed. The model is then deployed on multiple different datasets for PPGI task to assess its inference performance and generalization ability.
- The predicted values are analyzed by comparing them with ground truth measurements. The evaluation metrics, such as MAE, RMSE, and SNR to assess the model's effectiveness in practical settings.

The following Figure 1.1 serves as a general overview of the framework developed in this work.

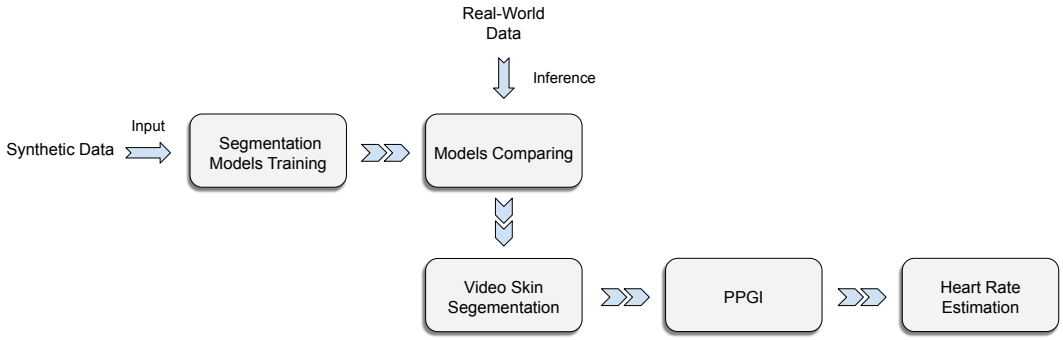


Figure 1.1.: Schematic structure of the framework.

1.3 Outline – waiting for correcting

This paper is structured as follows:

- Chapter 2: Background – Provides an overview of skin segmentation and its significance in computer vision applications, particularly in PPGI. It also discusses the advantages and challenges of using synthetic data for training deep learning models.
- Chapter 3: Related Work – Reviews PPG signal extraction techniques and heart rate estimation methods that are applied in this work, some previous state-of-the-art

deep learning architectures (CNN-based, attention-based, and backbone networks) research on segmentation, and reviews recent synthetic data applications.

- Chapter 4: Implementation – Describes details of the overall approach, including the experimental design and setup, dataset details, network architectures, and training configurations. It also explains the integration of the model into a video segmentation pipeline for PPGI applications.
- Chapter 5: Evaluation – Presents experimental results and discusses model performance based on various evaluation metrics. This chapter includes comparisons between different architectures, analysis of training strategies, generalization tests on real-world images, and validation in PPGI tasks.
- Chapter 6: Conclusion – Summarizes the key findings of the study, discusses the implications of using synthetic data for skin segmentation, and outlines potential directions for future research. Additionally, this work provides valuable insights into the applicability of synthetic data and deep learning-based skin segmentation models as a reference for future research in PPGI tasks.

2. Background

This chapter provides an overview of the foundational topics relevant to this study. Section 2.1 introduces Photoplethysmographic Imaging (PPGI), outlining its physiological basis, the transition from contact-based PPG to remote PPG (rPPG), evolution towards video-based PPGI systems, and fundamentals of signal processing for PPGI. Section 2.2 presents the role of skin segmentation in computer vision and its significance in enhancing signal extraction for physiological analysis. Section 2.3 discusses synthetic facial datasets, highlighting recent advancements in data generation techniques and their utility in training deep learning models. These components collectively establish the context for this thesis, with a particular emphasis on the application of synthetic data in skin segmentation and its integration within PPGI pipelines.

2.1 Photoplethysmographic Imaging (PPGI)

Photoplethysmographic Imaging (PPGI) is a non-contact technique for measuring physiological signals using optical imaging. It builds upon the principles of photoplethysmography (PPG), a well-established method for detecting blood volume changes in tissue through variations in light absorption [4].

2.1.1 PPG Principle

PPG was first developed in the 1930s [4], it is a widely used non-invasive optical technique for monitoring physiological parameters such as heart rate, oxygen saturation, and respiration rate [5, 6]. Figure 2.1 demonstrates the fundamental principle of PPG. It works by illuminating the skin and detecting variations in the intensity of reflected or transmitted light, which correlate with the cardiac cycle. The arterial blood component (depicted

in red in the diagram) produces the most significant modulation, creating the alternating current (AC) component of the PPG signal - this corresponds to the clear pulsatile waveform visible in the figure that synchronizes with the cardiac cycle. Meanwhile, the venous blood (shown in blue) and static tissues (represented in neutral tones) collectively establish the direct current (DC) baseline, seen as the steady background level upon which the pulsatile AC component is superimposed. The photodetector captures the composite light intensity fluctuations, yielding a raw signal that intrinsically combines the pulsatile AC component carrying cardiac information with a dominant DC offset from non-pulsatile elements. Through subsequent signal conditioning, including DC removal and bandpass filtering, the clinically relevant AC component emerges, revealing the characteristic PPG waveform morphology that enables the derivation of vital physiological parameters [5].

Traditional PPG systems rely on contact-based sensors placed on the skin (e.g., fingertip or earlobe), but these may be unsuitable in contexts requiring movement freedom or in cases of skin injury [7, 8]. This has led to the development of non-contact, video-based techniques.

2.1.2 rPPG

In contrast, remote photoplethysmography (rPPG) enables contactless measurement of blood volume pulse by analyzing subtle color changes in facial video frames captured using standard RGB cameras [9]. These color fluctuations are primarily caused by variations in blood perfusion and are most prominent in the green channel due to hemoglobin absorption characteristics [10].

The Image 2.2 depicts the fundamental optical principles underlying skin imaging, where a light source illuminates the skin surface and a camera sensor captures the resulting reflections. When light interacts with skin, it produces both specular reflection (surface glare, typically discarded) and diffuse reflection (subsurface scattering carrying physiological data). The light penetrates through distinct skin layers—epidermis, dermis, and hypodermis—with each layer modulating the light differently. The epidermis scatters shorter wavelengths and absorbs melanin, while the dermis reveals blood vessels and capillaries through hemoglobin's light absorption, enabling techniques like PPG [10].

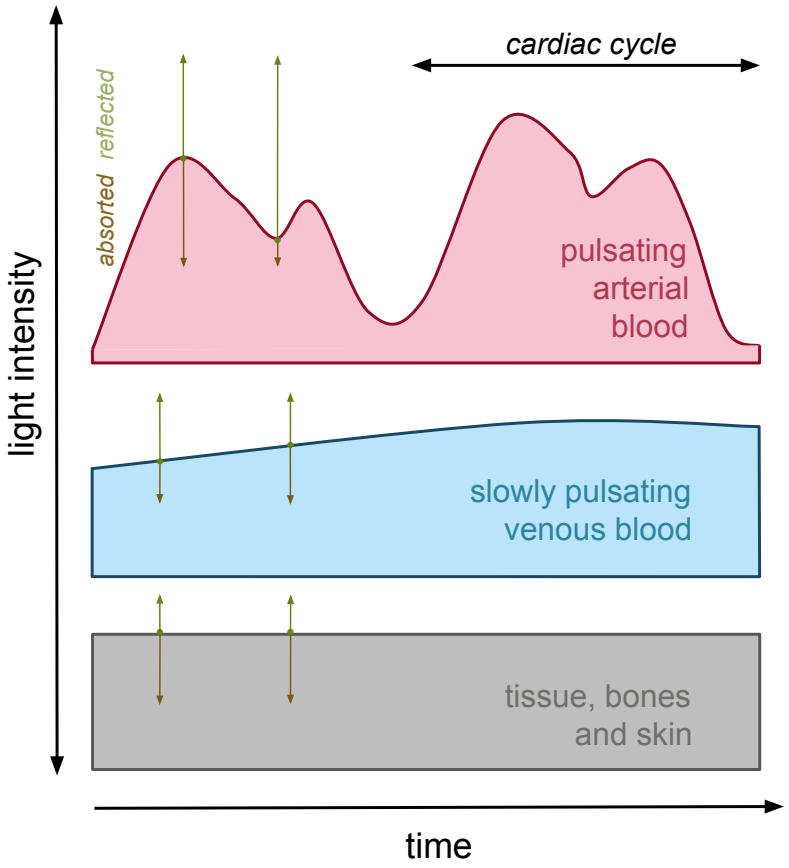


Figure 2.1.: Components of reflected light in the upper layers of human skin [7].

2.1.3 From rPPG to PPGI

While traditional rPPG methods typically rely on predefined or manually selected facial regions for signal extraction [9–11], PPGI refers to the broader framework that includes not only signal extraction but also upstream processes such as face detection, region-of-interest (ROI) selection, and downstream stages like signal enhancement and temporal filtering [12].

While promising for non-contact health monitoring, real-world PPGI applications face major challenges including motion artifacts [13], illumination variability [14], and low

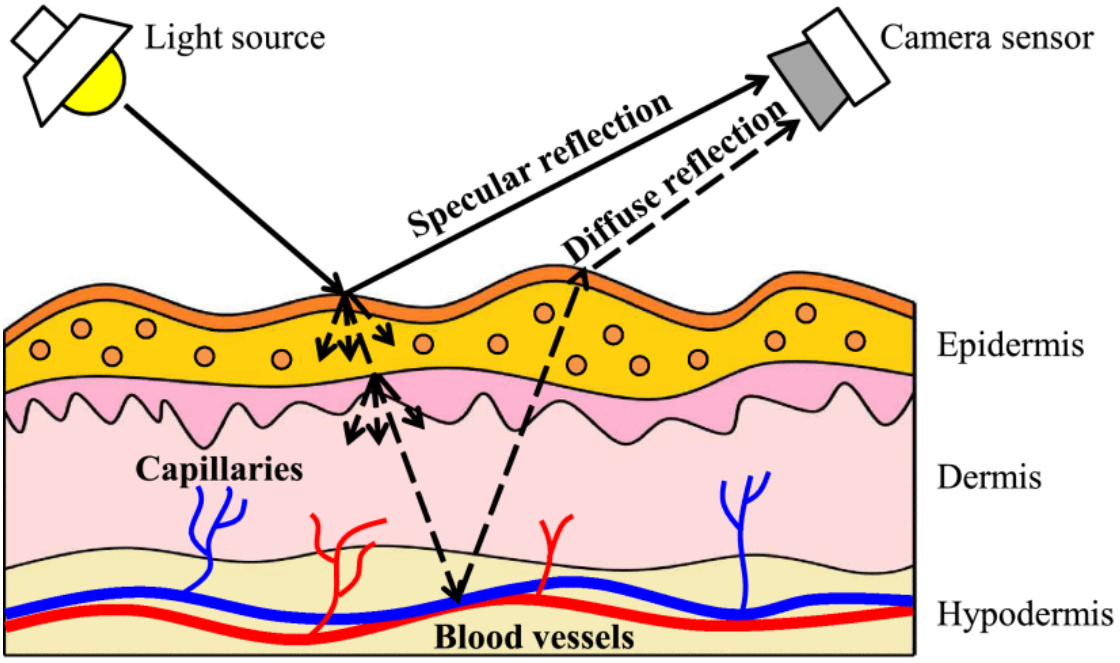


Figure 2.2.: Skin reflection model taken from [10].

signal-to-noise ratio (SNR) in uncontrolled environments [15].

A key factor affecting signal quality is the ROI selection. The signal is predominantly present in exposed skin regions such as the forehead and cheeks, but may degrade if non-skin areas like hair or background are included, especially when the subject is partially occluded or under non-uniform lighting [10, 11, 16].

To mitigate these issues, accurately identifying skin regions in each frame has been proposed as an effective strategy. A precise and adaptive skin segmentation approach can help isolate regions with reliable physiological signals while excluding irrelevant or noisy areas, thus improving the overall performance of the PPGI pipeline [3, 17, 18].

2.1.4 Signal Processing Fundamentals for PPGI

To ensure accurate heart rate estimation from extracted PPG signals, classical signal processing techniques such as bandpass filtering, Fast Fourier Transform (FFT), and peak

detection are commonly used in literature [9, 19].

Bandpass Filtering: Physiological signals related to heartbeats typically reside within a specific frequency band. Bandpass filters are applied to suppress irrelevant low-frequency trends (e.g., motion, illumination changes) and high-frequency noise [20]. This enhances the signal-to-noise ratio of the pulse component and ensures that further analysis focuses on relevant periodicities.

POS Algorithm: Wang *et al.* proposed algorithm POS (Plane-Orthogonal-to-Skin), which is a widely used method for robust extraction of pulse signals from facial videos by projecting the color signals onto a plane orthogonal to the skin tone [10]. The key insight is that motion and illumination changes tend to affect the RGB channels similarly, while the pulse-induced color variations differ in their projections. By removing the skin-tone component, POS enhances the pulse signal while suppressing common noise sources. The algorithm involves normalizing the RGB signals extracted from the face region, projecting them onto the orthogonal plane, and combining the projections using weighted sums. The resulting signal is then filtered and analyzed to estimate heart rate. POS has demonstrated strong robustness to motion artifacts and lighting changes compared to simpler methods. This work adopts the POS algorithm as the primary PPG signal extraction method in the PPGI pipeline.

The essential algorithmic framework of POS is summarized in Algorithm 1 [10] below:

Algorithm 1 Plane-Orthogonal-to-Skin (POS)

Require: A video sequence containing N frames

Initialize: $H = \text{zeros}(1, N)$, $l = 32$ (20 fps camera)

for $n = 1, 2, \ldots, N$ **do**

  $C(n) = [R(n), G(n), B(n)]^\top$ ▷ Spatial averaging

  if $m = n - l + 1 > 0$ **then**

    $C_n^i = \frac{C_{m-n}^i}{\mu(C_{m-n}^i)}$ ▷ Temporal normalization

    $S = \begin{pmatrix} 0 & 1 & -1 \\ -2 & 1 & 1 \end{pmatrix} \cdot C_n$ ▷ Projection

    $h = S_1 + \frac{\sigma(S_1)}{\sigma(S_2)} \cdot S_2$ ▷ Tuning

    $H_{m-n} = H_{m-n} + (h - \mu(h))$ ▷ Overlap-adding

  end if

end for

Ensure: The pulse-signal H

Fast Fourier Transform (FFT): FFT is a powerful algorithm that transforms time-domain signals into the frequency domain. When applied to rPPG signals, it helps identify the dominant frequency corresponding to the heart rate. Typically, the signal is first normalized and filtered, after which FFT is computed. The frequency with the highest magnitude in the physiological band is selected as the estimated pulse frequency [21]. This method is especially useful when the signal is relatively stable and periodic.

Peak Detection: Unlike FFT, peak detection methods operate directly in the time domain. After preprocessing (e.g., bandpass filtering and normalization), local maxima in the signal are identified as potential heartbeat peaks. The time intervals between successive peaks—known as RR intervals—are then used to compute heart rate using the formula:

$$HR = \frac{60}{\overline{RR}}, \tag{2.1}$$

where $\overline{RR}$ is the average time (in seconds) between detected peaks. This method is more sensitive to waveform morphology and can offer improved robustness in noisy or nonstationary conditions [16].

These methods constitute the fundamental processing blocks for HR estimation in most rPPG-based pipelines, including the one adopted in this work.

2.2 Skin Segmentation in Biomedical Applications

Skin segmentation refers to the process of identifying and isolating skin regions from digital images [22]. It serves as a foundational step in many computer vision and biomedical applications, including facial recognition [23], gesture analysis [24], emotion detection [25], and, notably, remote photoplethysmographic imaging (PPGI) [3]. As a prerequisite step in PPGI, skin segmentation aims to accurately localize skin regions to facilitate reliable physiological signal extraction. Inaccurate segmentation may introduce background noise and non-skin artifacts, which can impair signal quality and subsequent analysis, such as heart rate estimation [9, 16].

Early methods for skin segmentation relied heavily on color-based rules. By transforming images into color spaces such as RGB, HSV, or YCbCr, skin regions could be distinguished based on empirical thresholds [26–29]. These methods were simple and computationally efficient, and they performed adequately in controlled environments with uniform lighting and limited background variation. However, their robustness under real-world conditions was limited. Changes in illumination, differences in skin tone across ethnicities, and the

presence of shadows or occlusion often led to poor generalization and high false positive rates [22, 26].

To improve adaptability, statistical modeling techniques were introduced. Gaussian Mixture Models (GMMs), for example, learned probabilistic distributions of skin color in different color spaces, allowing for more flexible classification boundaries [28]. Bayesian classifiers and histogram-based models were also explored [22]. While these methods offered improvements, they still depended heavily on color information and were therefore vulnerable to lighting and camera differences [30].

As image analysis evolved, researchers began incorporating texture and spatial features to aid skin detection. Feature descriptors such as Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), and Gabor filters were employed to capture texture patterns associated with skin regions [22]. These features were combined with classical classifiers like Support Vector Machines (SVMs) or Random Forests. While this approach provided better robustness against background clutter and varying lighting, it often required careful feature engineering and extensive parameter tuning [22, 27].

The advent of deep learning marked a significant leap in the field. Convolutional Neural Networks (CNNs) [31] enabled end-to-end learning of hierarchical representations directly from data, eliminating the need for handcrafted features. Fully Convolutional Networks (FCNs) such as U-Net [32], originally proposed for biomedical image segmentation, became widely adopted due to their encoder–decoder structure, which balances local detail preservation with high-level semantic understanding.

More advanced architectures such as U-Net++ [33] and U-Net3+ [34] introduced nested skip connections and multi-scale fusion strategies to improve boundary localization and feature representation. These architectures have been proposed to improve feature representation and boundary localization, particularly for images with complex facial structure.

Despite their success, CNNs have limitations in modeling long-range dependencies, which can be important for capturing contextual cues in face images [35]. Recent models incorporate attention mechanisms or replace CNN components entirely with Transformers. Vision Transformers (ViTs) [36], for instance, model global relationships in an image via self-attention, enabling them to capture dependencies beyond the local receptive field. However, pure Transformer models often require large datasets and high computational cost [37].

To address these challenges, hierarchical Transformer architectures such as the Swin Transformer have been proposed [38]. The Swin Transformer introduces shifted window-based self-attention, which reduces computational complexity while capturing both local and

global image features efficiently. Building on this, hybrid models such as Swin-Unet [39] and SegNeXt [40] have emerged, combining the local feature extraction capabilities of CNNs with the global context modeling of Transformers. These architectures have demonstrated strong performance in medical and facial segmentation tasks, especially when applied to high-resolution inputs or images with occlusion and complex backgrounds.

While deep learning models have shown impressive results, their success often depends on the availability of large, annotated datasets [41]. For tasks such as facial skin segmentation, collecting pixel-level labels from real images is expensive and time-consuming. Synthetic face images, generated with consistent lighting, pose, and skin detail, offer a promising alternative [2, 42–44]. In this work, we explore the use of synthetic data to train a deep segmentation model and assess its effectiveness in a real-world biomedical application, specifically improving signal quality in PPGI. We aim to understand how well a model trained entirely on synthetic faces generalizes to real faces and whether this approach can reduce the reliance on real-world labeled datasets.

2.3 Synthetic Facial Datasets

In recent years, synthetic data has gained increasing attention in computer vision tasks, especially in facial analysis, where the availability of large-scale annotated datasets remains a key bottleneck [2, 43, 44]. One typical example is facial skin segmentation, which requires fine-grained pixel-level annotations that are extremely time-consuming and costly to obtain from real-world images [42]. Moreover, real datasets often suffer from imbalanced representation in terms of lighting conditions, head poses, occlusions, and demographic diversity, limiting the generalizability of trained models [2, 43].

To address these challenges, researchers have turned to synthetic face images as a promising alternative. Early efforts primarily relied on 3D morphable models (3DMM) to generate synthetic faces with controlled variations [45, 46]. While helpful, these approaches generally lacked realism in texture, expression variation, and lighting, which limited their applicability to downstream tasks such as segmentation [43, 46].

Recent advances in generative models—particularly Generative Adversarial Networks (GANs)—have significantly improved the realism and variability of synthetic face data [47]. GAN-based frameworks such as StyleGAN are capable of generating high-resolution facial images with lifelike texture, diverse expressions, and complex illumination settings [48, 49]. These properties make synthetic face datasets increasingly viable for training deep

learning models in tasks like skin segmentation, where high-quality pixel-level annotations are otherwise difficult to obtain.

Compared to real-world data, synthetic datasets offer distinct advantages: they are fully annotated by design, highly controllable in terms of attribute distribution (e.g., lighting, pose, occlusion), and can be generated in unlimited quantity at minimal cost [2, 44, 50]. However, a notable challenge remains—the domain gap between synthetic and real data, which may impact model performance when transferring to real-world scenarios [44, 50].

In this work, we investigate whether a deep learning-based skin segmentation model trained solely on synthetic face images can generalize effectively to real face videos within a PPGI (Photoplethysmographic Imaging) pipeline. This research focuses on evaluating the practical value of synthetic data in the context of challenging health-related computer vision tasks.

3. Related Work

This chapter outlines key research areas supporting this study. Section 3.1 presents PPG signal extraction and heart rate estimation, focusing on the signal processing approaches. Section 3.2 reviews deep learning networks for skin segmentation, including CNN-based and attention-based models, as well as commonly used backbones. Section 3.3 discusses the use of synthetic data in training segmentation networks. These topics together establish the technical basis for this work.

3.1 Signal Processing in PPGI

A standard PPGI pipeline (see Figure 3.1) typically consists of three stages: skin region (ROI) detection, PPG signal extraction, and heart rate (HR) estimation [9, 12]. Numerous works have explored different signal processing techniques at each stage to improve robustness against motion artifacts, illumination variation, and low signal-to-noise ratios [11, 16].

In the signal extraction stage, the Plane-Orthogonal-to-Skin (POS) method proposed by Wang *et al.* [10] has become a benchmark for rPPG/PPGI applications. By projecting the normalized RGB signals onto a chrominance plane orthogonal to the skin tone, POS enhances the pulsatile signal while suppressing common noise sources. Its simplicity and effectiveness have made it a foundational component in many subsequent PPGI pipelines [16, 51].

Once the PPG signal is extracted, heart rate (HR) estimation is typically performed either in the frequency domain or the time domain. Fast Fourier Transform (FFT) is the classical tool to transform the time-domain signal into the frequency domain, identifying dominant periodic components corresponding to the cardiac pulse [21]. Despite FFT's capability to provide a global frequency overview, it is sensitive to noise, spectral leakage,

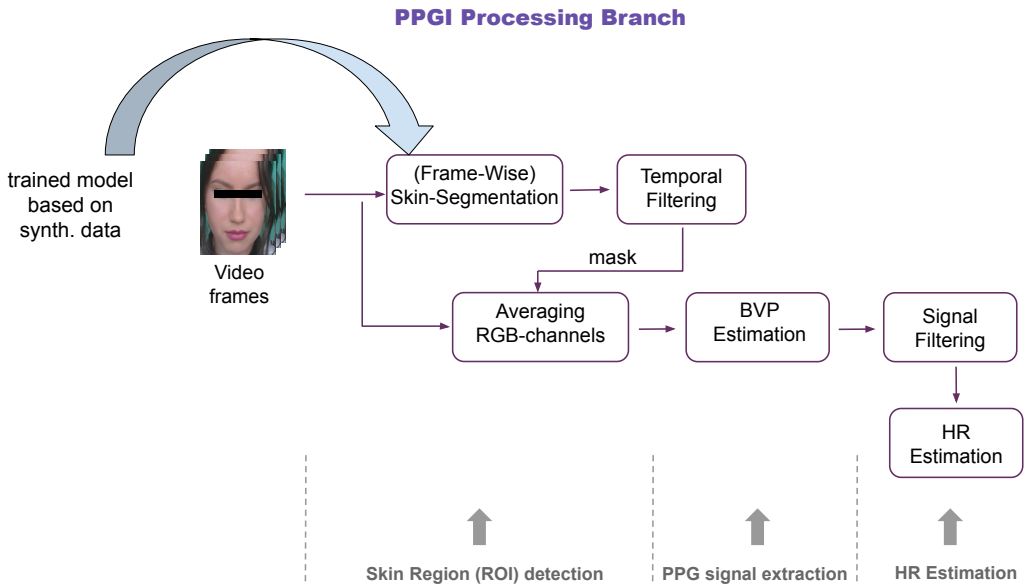


Figure 3.1.: Pipeline for PPG processing branch. It generally consists of stages: skin region (ROI) detection, PPG signal extraction, HR estimation.

and requires sufficiently long signal windows for reliable resolution. These limitations motivate complementary approaches or signal enhancement prior to FFT [52–54].

Complementary to frequency-domain analysis, time-domain peak detection estimates HR by identifying individual pulse peaks in the PPG waveform and calculating intervals between them. Peak detection provides high temporal resolution and can capture beat-to-beat variability, which FFT cannot [16, 55]. However, it is more vulnerable to local noise and baseline drifts, which can cause false or missed peaks [11, 16, 55]. Filtering and robust peak identification algorithms are critical to improve reliability.

Recent works have combined frequency and time domain methods to leverage their respective strengths. A typical approach involves using FFT to obtain a coarse HR estimate, which informs the design of a bandpass filter to isolate relevant frequencies before applying peak detection [56–58]. This hybrid scheme was also found to improve robustness against noise and motion artifacts in our evaluation (see Section 5.2.2).

Furthermore, the quality of the ROI—typically skin or facial areas—significantly affects

PPGI signal quality and HR estimation accuracy [10, 13]. Traditional methods commonly rely on heuristic or skin-color-based ROI selection [9, 11, 21, 59]. However, such approaches are sensitive to lighting variations, camera characteristics, and skin tone diversity, often resulting in incomplete or inaccurate skin region estimation, especially under challenging conditions. Several studies have shown that refining ROI by excluding non-skin regions such as hair, eyes, mouth, or shadowed areas improves the signal-to-noise ratio (SNR) and HR estimation performance [6, 13, 60]. These limitations motivate the use of more robust, learning-based skin segmentation models to achieve consistent and accurate ROI extraction.

In this study, we trained and compared several deep learning-based skin segmentation models based on synthetic face images. By accurately segmenting skin regions and masking out irrelevant areas before applying classical rPPG algorithms like POS, FFT, and peak detection, our method aims to enhance pulse signal extraction quality and robustness.

3.2 Deep Learning for Skin Segmentation

Skin segmentation has long been a foundational task in computer vision and biomedical imaging, enabling downstream applications such as facial analysis, emotion recognition, and remote photoplethysmographic imaging (PPGI) [3, 22–25]. Early methods relied on handcrafted rules and statistical models in color spaces [26–29], but the field has since transitioned toward data-driven deep learning approaches [41, 61].

The introduction of encoder–decoder frameworks such as U-Net [32] marked a pivotal moment in segmentation research. U-Net's symmetric structure and skip connections allowed precise localization of features, and it quickly became a standard for medical and skin segmentation tasks. To improve multi-scale learning and semantic detail, U-Net++ [33] introduced nested dense skip pathways, while U-Net3+ [34] proposed full-scale skip connections and deep supervision to enhance representation fusion.

Beyond the U-Net family, other convolutional models were explored. For example, DeepLabv3 [62] adopted atrous spatial pyramid pooling and proved effective in multi-class facial parsing tasks. BiSeNet[63] addressed real-time segmentation with a dual-branch structure to balance speed and accuracy, and was later extended in BiSeNetV2 for further efficiency[64].

Transformer-based models began to emerge more recently, offering improved modeling of long-range dependencies. TransUNet [35] was among the earliest attempts to incorporate

Vision Transformers (ViT) into the U-Net design, enhancing global context understanding for medical images. Swin-Unet [39] followed by replacing convolution with hierarchical Swin Transformer blocks, achieving strong performance in tasks requiring contextual awareness, such as face parsing under occlusion and lighting changes.

Parallel efforts also explored segmentation via mask prediction. MaskFormer [65] and its successor Mask2Former [66] unified semantic and instance segmentation via Transformer-based decoders with CNN backbones, offering a flexible and powerful alternative applicable to facial regions.

More recently, SegNeXt [40] demonstrated that efficient convolutional designs can remain competitive. By introducing multi-scale channel attention and optimizing for lightweight deployment (MSCAN-L backbone), SegNeXt offers a fast yet accurate option for real-time facial skin segmentation.

In addition to the above, notable models designed specifically for facial parsing include EHANet[67] and CE2P[68], which are tailored for fine-grained region discrimination in facial landmarks, lips, and skin. Though not directly used in our implementation, these architectures provide valuable insight into domain-specific design strategies.

In this work, we select six representative architectures for direct evaluation:

- CNN-based: U-Net, U-Net++, U-Net3+ with EfficientNetB0 as encoder;
- Transformer-based: Swin-Unet, Mask2Former(ResNet50 as backbone), and
- Hybrid convolutional: SegNeXt (MSCAN-L backbone).

These models represent the spectrum of modern segmentation architectures. We train each on synthetic face datasets and assess their performance in downstream PPGI scenarios. In addition, all selected architectures leverage encoder backbones initialized with pretrained weights—either from ImageNet [69] for classification-based backbones (e.g., EfficientNetB0, ResNet50), or ADE20K [70] for segmentation-specific variants (e.g., MSCAN-L). Pretraining on large-scale datasets provides generalizable low- and mid-level features that improve training stability, accelerate convergence, and often lead to better segmentation quality, particularly when downstream tasks (e.g., facial parsing or skin region isolation) involve subtle texture and boundary distinctions [70, 71]. The use of pretrained models is thus a practical and widely adopted strategy in modern segmentation pipelines, including ours. Unlike prior studies focused solely on segmentation accuracy, we evaluate how segmentation quality impacts physiological signal estimation, offering practical insights into their real-world utility.

3.3 Synthetic Data for Segmentation

In recent years, synthetic datasets have gained increasing attention in the field of semantic segmentation, particularly in scenarios where manual annotation of real images is expensive, time-consuming, or privacy-sensitive [2, 72, 73]. By generating highly detailed and pixel-wise labeled images, synthetic data provides a scalable and controllable alternative to real-world datasets [2, 74, 75].

Several benchmark synthetic datasets, such as SYNTHIA [72], GTA5 [73], and CARLA [76], have been widely used in urban scene understanding tasks. Models trained on these datasets have achieved competitive results in real-world benchmarks like Cityscapes [77] when combined with domain adaptation techniques [73, 76, 78].

In the domain of facial analysis and medical imaging, synthetic data is increasingly used for tasks such as facial part segmentation [2], skin lesion detection [79], and anatomical structure labeling [80]. One notable example is the work by Wood *et al.* [2], who introduced the Face Synthetics dataset—a large-scale collection of high-fidelity 3D rendered face images with dense semantic labels. See below for sample datasets (Figure 3.14) and an excerpt (Table 3.5) from their experimental results.

Figure 3.2.: Examples of generated and rendered synthetic faces that are randomly created to serve as training data. [2]

Table 3.1.: A comparison with the state of the art on the Helen dataset, using F1 score [2].

Method	Skin	Nose	Upper lip	Inner mouth	Lower lip	Brows	Eyes	Mouth	Overall
Guo *et al.* [81] AAAI'18	93.8	94.1	75.8	83.7	83.1	80.4	87.1	92.4	90.5
Wei *et al.* [82] TIP'19	95.6	95.2	80.0	86.7	86.4	82.6	89.0	93.6	91.6
Lin *et al.* [83] CVPR'19	94.5	95.6	79.6	86.7	89.8	83.1	89.6	95.0	92.4
Liu *et al.* [84] AAAI'20	94.9	95.8	83.7	89.1	91.4	83.5	89.8	96.1	93.1
Te *et al.* [85] ECCV'20	94.6	96.1	83.6	89.8	91.0	90.2	84.9	95.5	93.2
Wood *et al.* [2] (real)	95.1	94.7	81.6	87.0	88.9	81.5	87.6	94.8	91.6
Wood *et al.* [2] (synthetic)	95.1	94.5	82.3	89.1	89.9	83.5	87.3	95.1	92.0

Their study demonstrated that models trained on such synthetic data can generalize well to real images for tasks like landmark detection and face part segmentation [2]. It provides strong empirical support for the use of synthetic datasets in place of manually annotated real data, illustrating the performance gap narrowing between synthetic-trained and real-trained models.

In this work, we build upon the Face Synthetics dataset introduced by Wood *et al.* [2], and use it as the sole training source for a deep learning-based skin segmentation model. By leveraging its accurate and diverse annotations, we aim to evaluate whether a model trained purely on synthetic data can perform robustly when transferred to real-world physiological applications such as PPGI.

Other related works also highlight the value of synthetic data in segmentation tasks. For instance, Qiu *et al.* [86] proposed the SynFace dataset, a large-scale synthetic face dataset for training deep face recognition models. Their work demonstrated that synthetic-only training pipelines, when carefully constructed with identity-preserving generation and domain regularization techniques, can achieve competitive performance compared to models trained on real human face images. Similarly, Saxena *et al.* [87] synthesized annotated medical images to boost retinal vessel segmentation, eliminating the need for labor-intensive annotation processes.

Despite these promising outcomes, a key challenge in using synthetic data remains the domain gap—the visual and statistical discrepancy between synthetic and real-world data. This often leads to degraded performance in real-world deployment [74, 88]. A variety of domain adaptation strategies, including adversarial training (e.g., CyCADA [75]), image style translation (e.g., SimGAN [74]), and fine-tuning with limited real data, have been explored to address this issue.

Building on these studies, a broader consensus has emerged: for structured and visually consistent targets such as facial skin, segmentation models trained on synthetic images can

achieve performance that is competitive—even comparable—with those trained on real data [2, 86, 87]. This insight is particularly valuable in biomedical applications, where the availability of annotated real data is often limited by privacy or resource constraints [89].

In our study, we explore the direct impact of synthetic-data-trained skin segmentation models on downstream tasks, specifically PPGI - an area that remains underexplored in prior literature. Our work aims to investigate how well synthetic-data-trained models perform in real-world physiological signal extraction by assessing both segmentation quality and signal-level metrics such as heart rate accuracy and signal-to-noise ratio.

4. Implementation

This chapter presents the detailed implementation of our study. Figure 1.1 in Section 1.2 provides an overview of the workflow. We structure our method into two key stages for clarity (model training branch and PPG processing branch). The model training branch focuses on the development and evaluation of a deep learning-based segmentation model. It includes a description of the network architectures used, the datasets employed, and the experimental setup. The PPG processing branch investigates the impact of segmentation quality on PPG signal extraction, which includes a description of the datasets used and the experimental setups applied.

4.1 Model Training Branch

As shown in Figure 4.1, this branch begins by exploring and comparing the performance of several state-of-the-art deep learning models trained on synthetic face images for the task of skin segmentation. This section introduces the implementation details in this branch of this study, including: dataset details and corresponding experimental setup.

4.1.1 Networks Building

The skin segmentation models used in this work can be grouped into three categories:

CNN-based: U-Net, Net++, U-Net3+

U-Net [32], U-Net++ [33], and U-Net3+ [34] were implemented, each using EfficientNet-B0 as the encoder. Their architectures are illustrated in Figure 4.2. Compared to the original U-Net, U-Net++ incorporates nested skip connections to enhance feature propagation, while U-Net3+ adopts full-scale skip connections for multi-scale feature fusion.

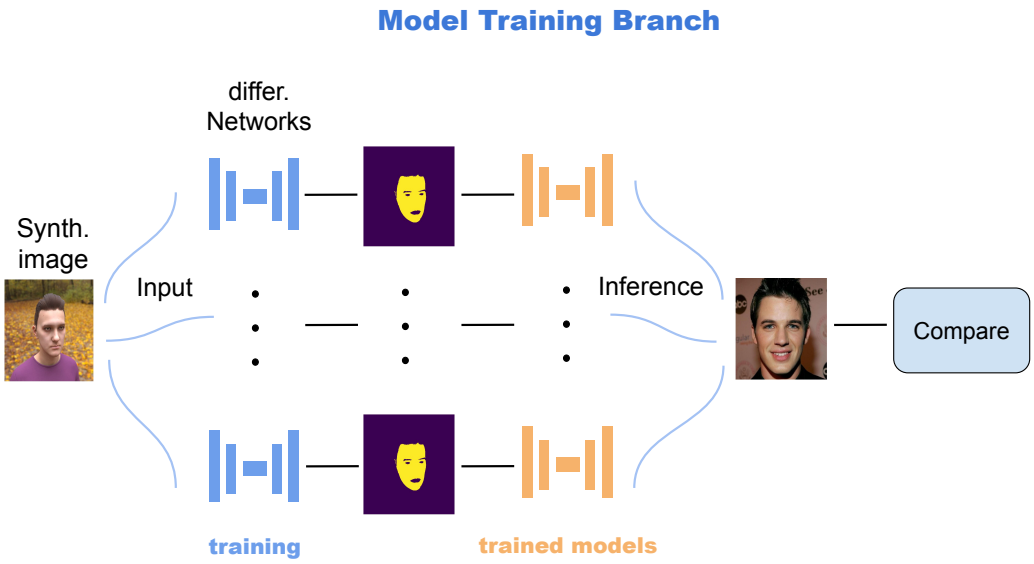


Figure 4.1.: Workflow for model training branch

All three networks share the same encoder: EfficientNet-B0, a compact and efficient CNN introduced by Google Brain [90]. EfficientNet-B0 serves as the baseline of the EfficientNet family, designed via Neural Architecture Search (NAS) [91] to achieve optimal accuracy-efficiency trade-offs. The architecture applies compound scaling to uniformly scale depth, width, and resolution. This study adopts only the B0 variant; structural differences with larger variants (B1–B7) are summarized in Appendix B.2. The detailed structure of EfficientNet-B0 is listed in Table 4.1.

The MBConv blocks follow the MobileNetV2-style inverted bottleneck design [91], and SE refers to the Squeeze-and-Excitation mechanism [92] used to enhance channel-wise attention.

Transformer-based Models: Swin-Unet and Mask2Former

In this work, we implemented two Transformer-based segmentation models: Swin-Unet [39], commonly used in medical image segmentation, and Mask2Former [66], a

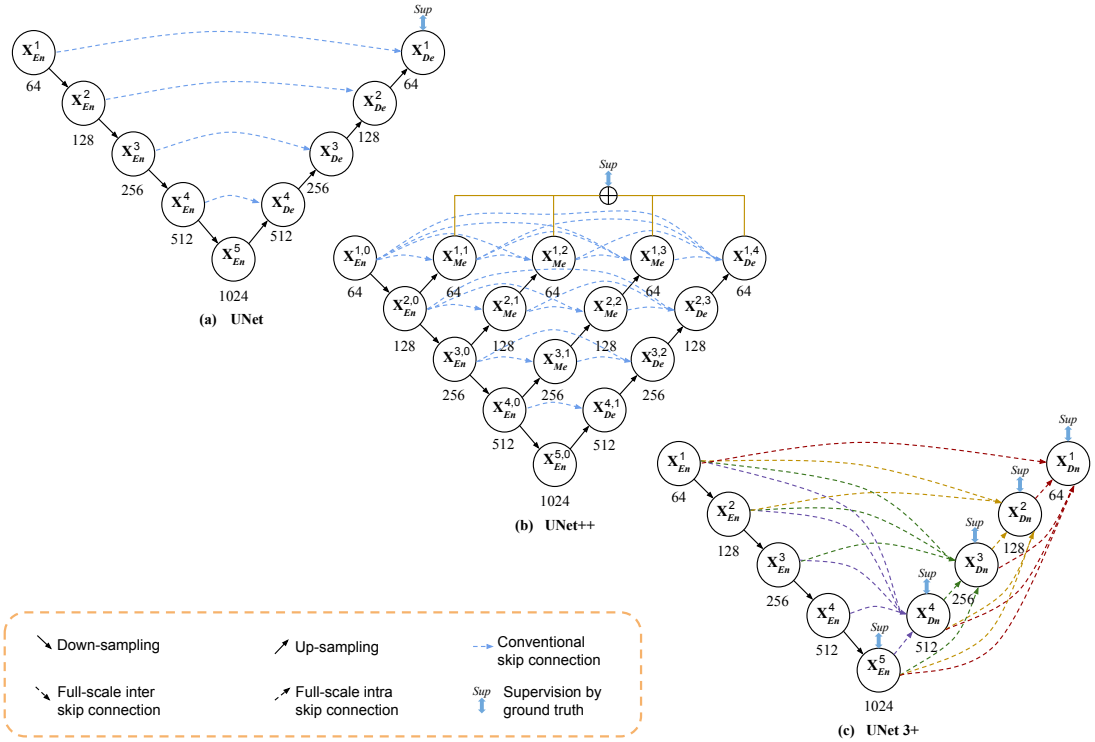


Figure 4.2.: Architectures of U-Net (a), U-Net++ (b), and U-Net3+ (c). Numbers inside circles indicate the depth of each node [34].

general-purpose segmentation architecture. These two models represent different design paradigms: hierarchical window-based attention vs. DETR-style masked attention.

The Swin-Unet architecture (Figure 4.3) is based on Swin Transformer blocks and follows a symmetric encoder-bottleneck-decoder design, similar to U-Net. All components are built using hierarchical Swin Transformer blocks, allowing efficient local-global feature modeling while maintaining spatial resolution.

In our implementation, we used the Swin Transformer-Tiny variant, whose structural specifications are shown in Table 4.2. More detailed comparisons between the Tiny, Small, Base, and Large variants are provided in Appendix B.4.

The key component, shifted window multi-head self-attention (SW-MSA), enables efficient contextual aggregation with reduced computational cost by limiting attention within

Table 4.1.: EfficientNet-B0 architecture [90]. MBConvN: Inverted bottleneck block with expansion factor N. SE=0.25: Squeeze-and-Excitation ratio. k$n\times n$: Kernel size in depthwise convolutions.

Stage	Operator	Resolution	#Channels	#Layers
1	Conv3x3	224×224	32	1
2	MBConv1, k3x3	112×112	16	1
3	MBConv6, k3x3	112×112	24	2
4	MBConv6, k5x5	56×56	40	2
5	MBConv6, k3x3	28×28	80	3
6	MBConv6, k5x5	14×14	112	3
7	MBConv6, k5x5	14×14	192	4
8	MBConv6, k3x3	7×7	320	1
9	Conv1x1 + Pooling + FC	7×7	1280	1

Table 4.2.: Structure of Swin Transformer-Tiny [38]

Stage	**Resolution**	**Channels**	**Swin Blocks**	**Window Size**
1	56×56	96	$2 \times$ (W-MSA + SW-MSA)	7×7
2	28×28	192	$2 \times$ (W-MSA + SW-MSA)	7×7
3	14×14	384	$6 \times$ (W-MSA + SW-MSA)	7×7
4	7×7	768	$2 \times$ (W-MSA + SW-MSA)	7×7

non-overlapping windows and introducing cross-window connections between successive layers.

Mask2Former [66] builds upon the MaskFormer [65] and DETR [93] frameworks. It integrates a CNN backbone, a pixel decoder, and a Transformer decoder into a unified segmentation system, as illustrated in Figure 4.4.

In this architecture, the CNN backbone (ResNet-50 [94], whose structure is summarized in Table 4.3) extracts multi-scale features, which are then refined by the pixel decoder. These features are further processed by the Transformer decoder, which utilizes a set of learnable queries. Through masked attention, each query attends to specific image regions, enabling class-aware and spatially-aware mask prediction. This design improves segmentation accuracy while maintaining inference efficiency. Architectural comparisons between MaskFormer and Mask2Former are provided in Appendix B.1. ResNet has multiple versions with varying depths (see Appendix B.3), among which ResNet-50 and ResNet-101 are the most commonly used.

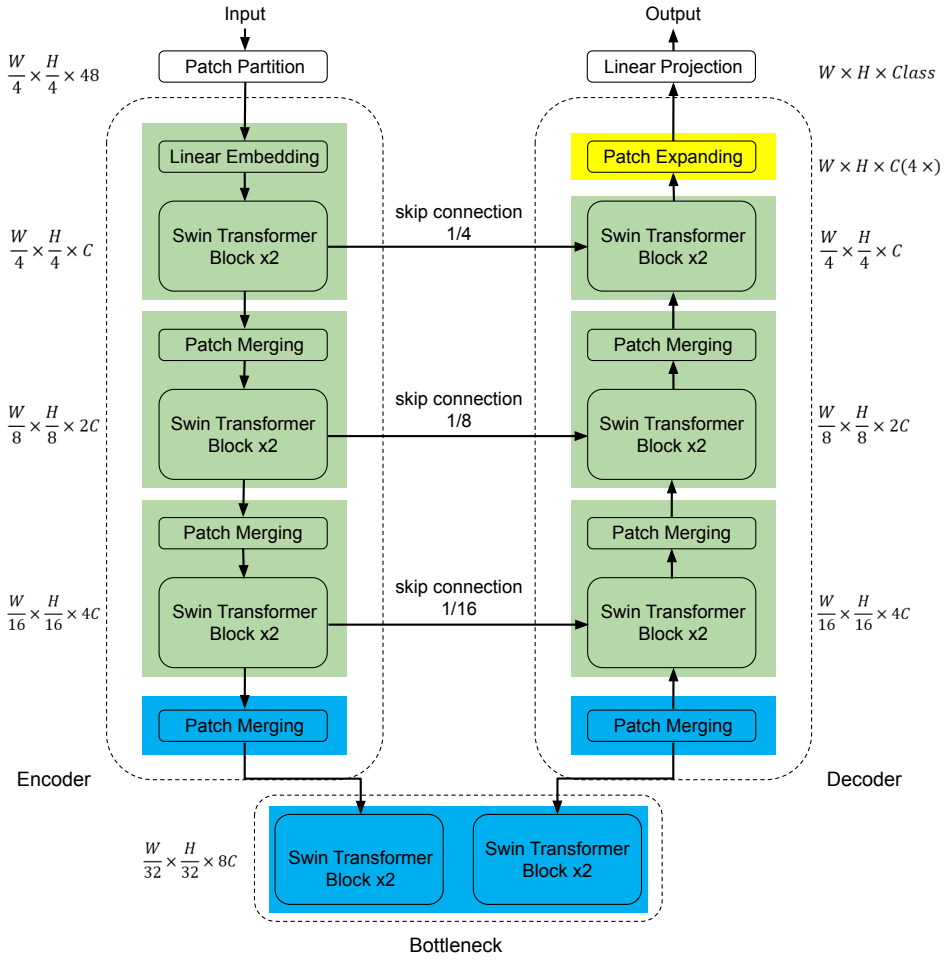


Figure 4.3.: Overall architecture of Swin-UNet. The model adopts an encoder–decoder structure with skip connections, constructed entirely from Swin Transformer blocks [39].

Hybrid Convolutional Model: SegNeXt

We also implemented SegNeXt [40], a recent hybrid convolutional architecture that rethinks attention mechanisms in semantic segmentation. It adopts an encoder-decoder framework and combines efficient multi-scale convolutional modules with a lightweight global context decoder.

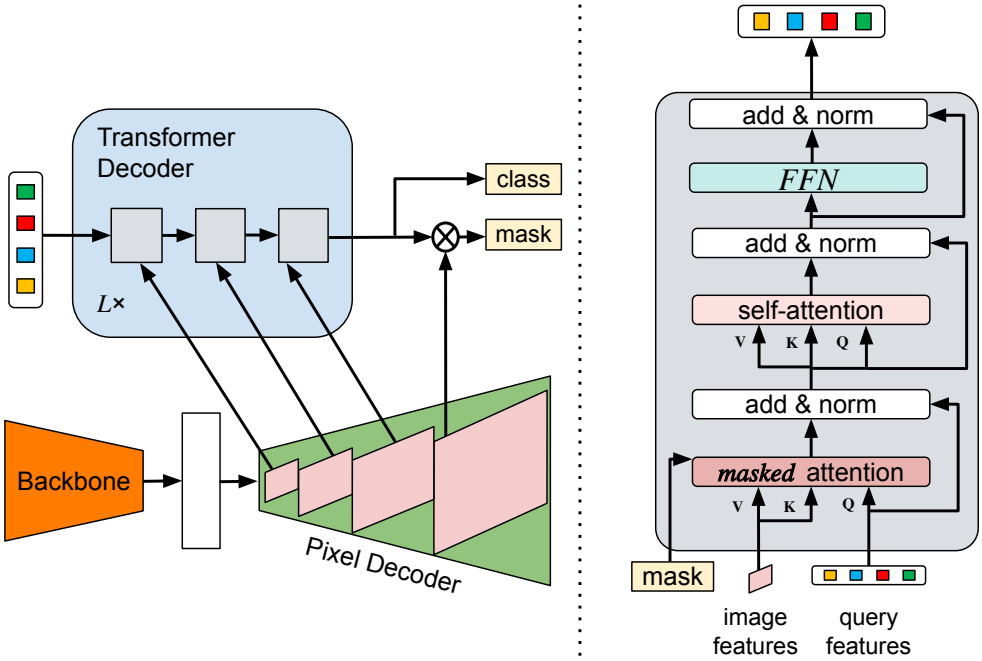


Figure 4.4.: Overall architecture of Mask2Former [66].

Table 4.3.: ResNet-50 Network Architecture. It includes 50 layers with trainable weights (including convolutional and fully connected layers), and is mainly divided into 5 stages (Stage 0 to Stage 4)

Stage	Layer Name	Output Size	Block Type	Repeats
Stage 0	Conv1 + MaxPool	112×112	–	–
Stage 1	Conv2_x	56×56	Bottleneck	3
Stage 2	Conv3_x	28×28	Bottleneck	4
Stage 3	Conv4_x	14×14	Bottleneck	6
Stage 4	Conv5_x	7×7	Bottleneck	3
Global AvgPool + FC		1×1	–	–

The encoder uses the Multi-Scale Convolutional Attention Network (MSCAN), which

replaces conventional self-attention with a Multi-Scale Convolutional Attention (MSCA) module. As illustrated in Figure 4.5, each MSCA block consists of:

- A depthwise convolution for local feature aggregation;
- Strip convolutions with kernel sizes 7, 11, and 21, implemented using sequential $1 \times k$ and $k \times 1$ convolutions to efficiently model elongated and multi-scale structures;
- A 1×1 convolution to generate spatial attention weights.

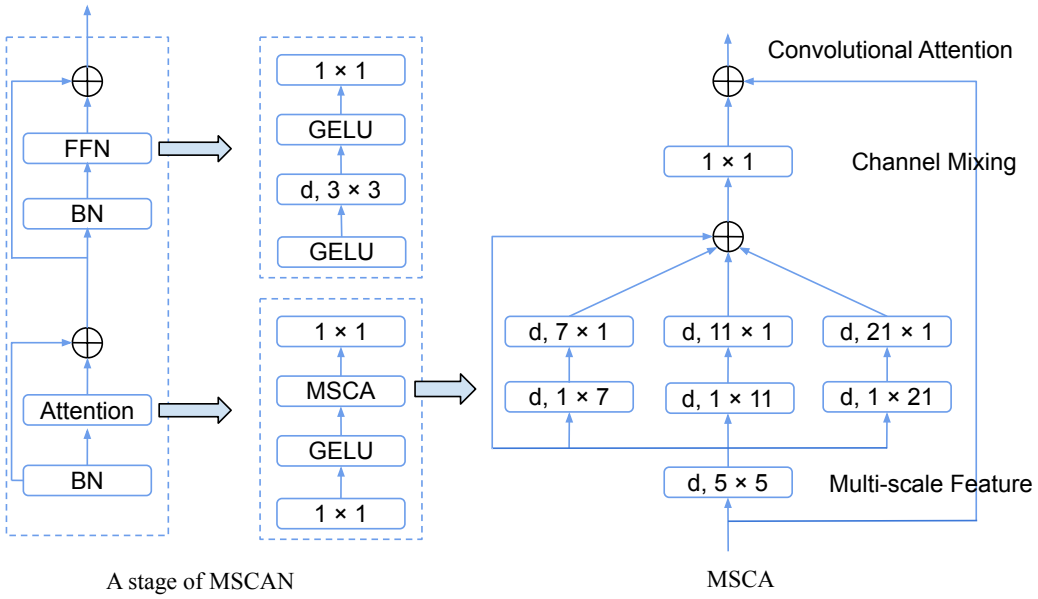


Figure 4.5.: Architecture of MSCA and MSCAN [40]. $(d, k_1 \times k_2)$ indicates a depthwise convolution with kernel size $k_1 \times k_2$.

The encoder is hierarchically organized into four stages, each starting with a downsampling layer (3×3 convolution, stride 2), followed by MSCA blocks. The authors introduced four model variants: MSCAN-T, MSCAN-S, MSCAN-B, and MSCAN-L. In our implementation, we adopt MSCAN-L as the backbone. A comparison of these variants is provided in Appendix B.5.

The decoder uses the lightweight *Hamburger module* (Figure 4.6), which aggregates features from Stages 2 to 4. Stage 1 is omitted due to its high computational cost and low semantic value. Unlike MLP-based or CNN-based decoders, the Hamburger

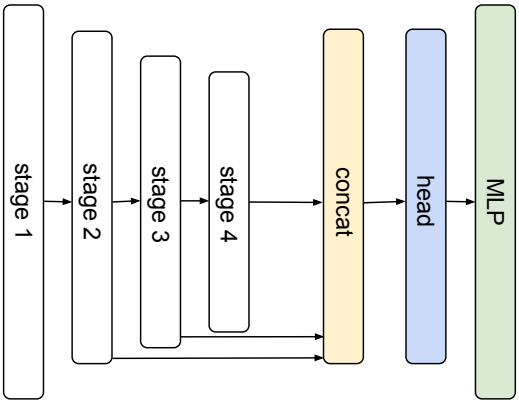


Figure 4.6.: Lightweight Hamburger structure used in SegNeXt [40].

module efficiently captures global context with minimal complexity. All models in this study are implemented using the Segmentation Models PyTorch library [95] and the MMSegmentation framework [96].

4.1.2 Dataset

Training Data: The training data for model training in this study is the Face Synthetics dataset, which was provided by Microsoft [2], a collection of diverse synthetic face images with ground truth labels. Figure 4.7 is the visualization of some samples.

The dataset contains: 100,000 images of faces at 512×512 pixel resolution; 70 standard facial landmark annotations; per-pixel semantic class annotations. Each pixel in the segmentation image is assigned one of the class labels, the label-to-class mapping is summarized in Table 4.4.

However, during our analysis of the training data, we identified the absence of the `FACEWEAR` (18) label in the dataset. The detailed analysis is presented in the Appendix A.1.

Figure 4.7.: Dataset samples [2].

4.1.3 Experimental Setup

Data Preprocessing: To further improve the model's generalization capability across real-world scenarios with varying image and video frame quality, we implemented an extended set of data augmentation strategies. In addition to standard techniques, including random zooming, rotation, brightness/contrast adjustment, and noise injection, we incorporated random grayscale conversion and motion blur simulation. Additionally, to improve computational efficiency, we reduced the input resolution from 512×512 to 256×256 pixels using local averaging downsampling. Due to the constraints of the pre-trained model and the window size requirements of Swin-Unet, the input data was further cropped to 224×224. Subsequently, we divided the synthetic dataset into training, validation, and test sets at a 6:2:2 ratio randomly, while using the entire real-world dataset for comparative inference.

Furthermore, we compared two distinct training strategies: Multi-Labels vs. Binary Labels. In the Multi-Labels experiment, the original 19-class labels from the dataset mask were preserved during training. In contrast, the Binary Labels experiment involved remapping the labels during preprocessing, where regions originally labeled as `SKIN` and `NOSE` were merged and assigned to `SKIN = 1`, while all other regions were remapped to `NON_SKIN = 0`. Regions labeled as `255` were consistently ignored during training. The remapped label-to-class relation is shown in Table 4.5.

For PPG signal extraction, we only considered the regions that are predicted as the label of `SKIN`. Experimental results (see Evaluation 5.1.2) demonstrate that training with

Table 4.4.: Label definitions used in segmentation.

Class	Label ID	Class	Label ID
BACKGROUND	0	HAIR	13
SKIN	1	BEARD	14
NOSE	2	CLOTHING	15
RIGHT_EYE	3	GLASSES	16
LEFT_EYE	4	HEADWEAR	17
RIGHT_BROW	5	FACEWEAR	18
LEFT_BROW	6	IGNORE	255
RIGHT_EAR	7		
LEFT_EAR	8		
MOUTH_INTERIOR	9		
TOP_LIP	10		
BOTTOM_LIP	11		
NECK	12		

Table 4.5.: Label remapping scheme for binary skin segmentation.

Original Label ID(s)	Remapped Class
1, 2	SKIN (1)
0, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18	NON-SKIN (0)
255	IGNORE (255)

remapped labels significantly enhances model performance.

Implementation Details: In this study, we implemented multiple deep learning architectures (follow the network building as introduced in Section 4.1.1) and comparatively trained them on synthetic data. Table 4.6 presents a comprehensive overview of the following state-of-the-art models chosen for comparison, detailing their respective hyperparameter settings and pretraining methodologies. Model initialization utilized ImageNet- and ADE20k-pretrained weights obtained from the Segmentation-Models-PyTorch [95] and MMSegmentation [96] libraries. Subsequent fine-tuning was performed with completely unfrozen network parameters within these frameworks.

All models were trained on NVIDIA A100-PCIE-40GB GPU. Specifically, the Mask2Former was trained using 4 GPUs with a batch size of 16 for each in a distributed manner, while

the remaining models were trained on a single GPU with a batch size of 64. Training was stopped when no further improvement in performance was observed on the validation images.

Table 4.6.: Overview of applied model architectures and training specifications, detailing encoder types (see Section 4.1.1), presence of attention (Att.), loss functions, optimizer, initial learning rate (LR), and utilized pretraining dataset types.

Model details			**Training details***			**Pretraining datasets**	
Model	**Encoder**	**Att.**	**Loss**	**Optimizer**	**LR**	**ImageNet**	**ADE20k**
UNet [32]	EfficientNetB0 [90]	✗	BCE+Dice	AdamW	2e-4	✓	✗
UNet++ [33]	EfficientNetB0 [90]	✗	BCE+Dice	AdamW	2e-4	✓	✗
UNet3+ [34]	EfficientNetB0 [90]	✗	BCE+Dice	AdamW	2e-4	✓	✗
Swin-Unet [39]	Swin-Transformer-Tiny [38]	✓	BCE+Dice	AdamW	2e-4	✓	✗
Mask2Former [66]	ResNet50 [94]	✓	CE+Dice	AdamW	1e-4	✗	✓
SegNeXt [40]	MSCAN-L [40]	✓	BCE+Dice	AdamW	6e-5	✗	✓

* BCE is an abbreviation for binary cross-entropy loss, CE corresponds to cross-entropy loss. Initializations (Loss, Optimizer, LR) are adopted from the MMSegmentation [96] library for Mask2former and SegNeXt, the remaining models employ manually tuned configurations.

4.2 PPG Processing Branch

PPGI pipeline (see Figure 3.1 in Section 3.1). It illuminates that he best-performing model after the previous branch is integrated into a conventional PPGI pipeline and applied to inference on several publicly available video datasets. Finally, the heart rates estimated from the experiments are compared with the ground truth values to evaluate the feasibility of using synthetic face images to train deep learning models and extend their application to real-world scenarios. This section introduces the datasets commonly used in iPPG research and the implementation details of the PPG processing branch.

4.2.1 Dataset

UBFC dataset: UBFC-rPPG (stands for Univ. Bourgogne Franche-Comté Remote Photo-PlethysmoGraphy) [97] dataset was constructed using a custom-developed C++ application for video capture, employing an affordable Logitech C920 HD Pro webcam. The videos were recorded at 30 frames per second with a resolution of 640×480 pixels, stored

in uncompressed 8-bit RGB format. For ground truth PPG data acquisition, a CMS50E transmissive pulse oximeter was synchronized with the video recordings to capture both the PPG waveform and corresponding heart rate measurements.

During data collection, participants were seated approximately one meter away from the camera, ensuring full facial visibility throughout the recordings. All experiments were conducted in indoor environments under varying lighting conditions, including natural sunlight and artificial illumination, to simulate real-world scenarios.

This setup provides a standardized benchmark for evaluating remote photoplethysmography (rPPG) algorithms under controlled yet diverse illumination conditions.

PURE dataset: The PURE (Pulse Rate Detection) dataset [98] comprises recordings of 10 individuals (8 male, 2 female), each participating in six distinct controlled head motion scenarios, resulting in a total of 60 one-minute video sequences. During the recordings, both facial video data and reference pulse signals were collected simultaneously.

The video sequences were captured using an eco274CVGE camera from SVS-Vistek GmbH at a frame rate of 30 Hz and a cropped resolution of 640×480 pixels, using a 4.8 mm lens. Simultaneously, ground truth pulse waveforms and SpO_2 values were acquired using a Pulox CMS50E finger clip pulse oximeter, which samples at 60 Hz. Test subjects were positioned at an average distance of 1.1 m from the camera. Illumination was provided by natural daylight through a large window positioned in front of the subject's face, with moderate changes in brightness due to varying cloud cover.

Each subject was recorded under six predefined conditions designed to simulate different levels and types of head motion:

- Steady: The subject sits still, looking directly at the camera while avoiding any head motion.
- Talking: The subject simulates a video call by speaking naturally, while trying to minimize head movement.
- Slow Translation: The subject moves their head horizontally in sync with a moving rectangle displayed on a screen. The rectangle moves at an average speed of 7% of the face height per second (average face height $\approx$ 100 pixels). This setup ensures repeatable side-to-side motion parallel to the image plane.
- fast Translation: Similar to the slow translation scenario, but with the target rectangle moving at twice the speed.

- Small Rotation: Multiple visual targets are placed in a circle (radius 35 cm) around the camera. Subjects are instructed to follow these targets in a predefined sequence, rotating their heads (not just eyes). The average head rotation angle is approximately 20°, varying slightly with subject-camera distance (1–1.3 m).
- Medium Rotation: This scenario follows the same procedure as the small rotation condition but with targets placed 70 cm from the camera, increasing the average head rotation angle to around 35°.

All recordings were performed while subjects were at rest. The recorded pulse rates vary both across and within subjects, with values ranging from a minimum of 42 BPM to a maximum of 148 BPM, as measured by the oximeter.

KISMED PPGI Dataset: This dataset was collected at the KISMED Institute of TU Darmstadt under a controlled experimental setup designed to evaluate facial video-based physiological signal estimation under various real-world conditions. The recordings were conducted using two types of cameras: a Logitech C930c USB webcam (recording at 640×480 pixels using the YUV422 format) and a SIGMA fp mirrorless full-frame camera (recording at 3840×2160 pixels in lossless CinemaDNG format). Both cameras were mounted at the eye level of the subject and positioned at a standard distance of 1 meter. The dataset also includes synchronized BVP signals recorded via a CONTEC CMS50E finger pulse oximeter, attached to either index finger. Synchronization between the BVP signal and video data was achieved using UNIX timestamps and an optical flash signal triggered by an Arduino Uno at the beginning and end of each recording.

The dataset includes a variety of recording scenarios designed to simulate diverse and challenging real-world conditions for facial video-based physiological monitoring.

- Lighting variations: The subject remains seated at a fixed distance from the camera while the ceiling light conditions vary across three sub-scenarios: always on, always off, and rhythmically toggled on/off to simulate different illumination environments.
- Uneven facial illumination: A secondary halogen floodlight is positioned laterally to cast shadows on one side of the face, introducing asymmetric lighting. In an extended version of this scenario, the ceiling light is turned off, and only the side lighting is used. The halogen lamp alternates between on and off every 20 seconds. Videos for this scenario last 120 seconds.
- Varying camera distance: The subject alters their distance from the camera during recording, starting at 1 meter and increasing the distance by 0.5 meters every 30

seconds until reaching 3 meters, then returning to 1 meter in the same stepwise manner. The duration of videos in this scenario is 270 seconds.

- Recording during physical movement: The subject performs squats while maintaining gaze toward the camera. This setup is used to evaluate the effect of motion artifacts and changes in facial alignment.
- Facial occlusion variations: The study includes participants with diverse types of facial coverings such as makeup, scarves, and eyeglasses to examine the impact of occlusions.
- Changes in facial orientation: The subject is instructed via a graphical user interface to look at predefined markers in the room, causing intentional translational and rotational head movements. For translation tasks, the subject stands upright, and the camera height is adjusted to maintain a frontal view

4.2.2 Experimental Setup

As depicted in Figure 1.1, after evaluating the performance of various segmentation networks in the Model Training Branch, the best-performing model is selected for this stage. For each RGB video frame from different datasets, a face-centered crop with a resolution of 256×256 is applied. This preprocessing step ensures consistency with the training data, as the model was trained on images of size 256×256, and enables optimal inference performance by focusing on the facial region. Subsequently, frame-wise skin segmentation is performed using the segmentation model. Next, temporal filtering is applied to the binary masks by performing a pixel-wise logical AND operation over a sliding window of length k. These temporally filtered masks are then used to extract skin pixels, which are averaged per color channel for each frame to form an RGB signal. Then the PPGI signal is estimated from the RGB signal using the robust and commonly adopted POS method [10]. Finally, we calculate the predicted heart rate using the extracted PPGI signal and compare it against the ground truth value of the physiological signal provided by the dataset to validate the effectiveness of our model.

5. Evaluation

This chapter presents a comprehensive evaluation of the proposed skin segmentation model and its performance on PPGI tasks. The evaluation is divided into two main parts. The first part assesses the segmentation model's performance using standard semantic segmentation metrics on the selected datasets. The second part investigates how different segmentation outcomes affect the quality of the extracted PPG signals.

For both parts, we first describe the evaluation setup, including datasets, metrics, and implementation details. Then, we provide quantitative results along with qualitative analyses to highlight the strengths and limitations of the model in both standalone segmentation tasks and downstream physiological signal extraction.

5.1 Segmentation Model Evaluation

This section focuses on evaluating the performance of the skin segmentation models introduced in Section 4.1.1. To verify the effectiveness and generalizability of the models, we design a multi-step evaluation procedure covering both segmentation accuracy and downstream task utility. First, we describe the evaluation setup, including datasets, preprocessing and the evaluation metrics used, to ensure fair and reproducible benchmarking. Then, we assess segmentation performance using widely adopted quantitative metrics and qualitative visualizations. This step helps verify whether the models can accurately extract skin regions under varying visual conditions.

5.1.1 Evaluation Setup

Inference Data: Since the CelebAMask-HQ [99] dataset is derived from real facial images, these samples are employed in the model's inference stage in this study and will be

compared with inference results from synthetic data to examine the generalizability of models trained on artificially synthesized facial data when applied to real-world scenarios.

The CelebAMask-HQ is a large-scale face image dataset that has 30,000 high-resolution face images selected from the CelebA dataset by following CelebA-HQ. Each image has a segmentation mask of facial attributes corresponding to CelebA. The masks of CelebAMask-HQ were manually annotated with the size of 512 $\times$ 512 and 19 classes, including all facial components and accessories such as skin, nose, eyes, eyebrows, ears, mouth, lip, hair, hat, eyeglass, earring, necklace, neck, and cloth. Sample seen in Figure 5.1.

Notably, the class assignments for `SKIN`(label = 1) and `NOSE`(label = 2) (both corresponding to skin regions used for PPG signal extraction in this experiment) remain consistent across the two datasets. Furthermore, the total number of categories is identical in both datasets, ensuring controlled experimental variables.

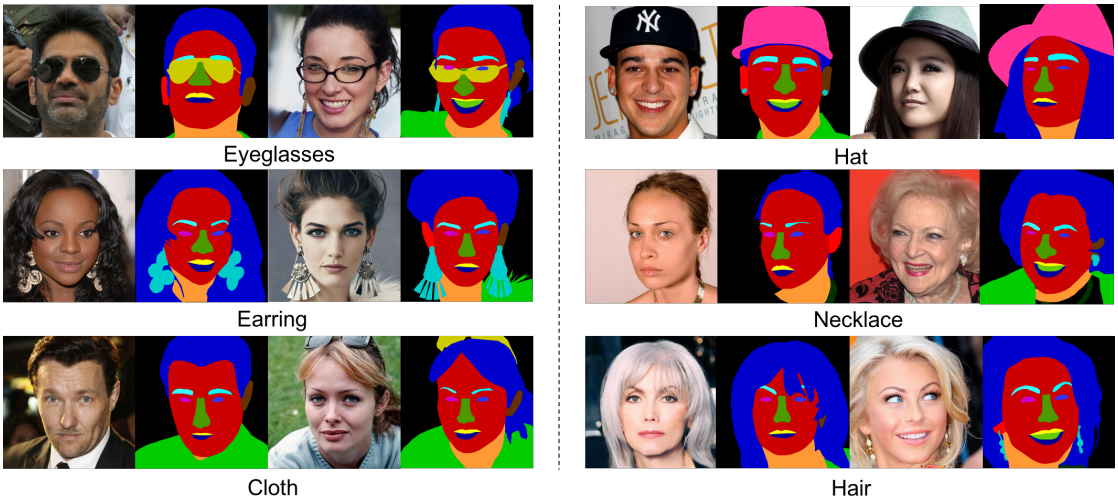


Figure 5.1.: Data samples of CelebAMask-HQ [99].

Data Preprocessing: To ensure consistency with the training data, the evaluation datasets are downsampled to a resolution of 256$\times$256 (or 224$\times$224 for Swin-Unet) using the same preprocessing strategy.

Inference is conducted on the same GPU model used during training, where each test image is processed individually with a batch size of 1.

Evaluation Metrics:

- Pixel Accuracy (PA):

$$PA = \frac{\sum_i TP_i}{\sum_i (TP_i + FP_i + FN_i)} \tag{5.1}$$

  This metric measures the overall percentage of correctly predicted pixels across all classes. It provides a general indication of segmentation accuracy.

- Mean Intersection over Union for skin class ($\text{mIoU}_{\text{skin}}$):

$$\text{mIoU}_{\text{skin}} = \frac{TP_{\text{skin}}}{TP_{\text{skin}} + FP_{\text{skin}} + FN_{\text{skin}}} \tag{5.2}$$

  This evaluates the overlap between the predicted skin region and the ground truth, and is particularly important in our context, where skin segmentation quality directly affects downstream tasks.

- Dice Coefficient for skin class ($\text{Dice}_{\text{skin}}$):

$$\text{Dice}_{\text{skin}} = \frac{2 \cdot TP_{\text{skin}}}{2 \cdot TP_{\text{skin}} + FP_{\text{skin}} + FN_{\text{skin}}} \tag{5.3}$$

  Dice is a commonly used similarity measure for segmentation tasks. It emphasizes the agreement between predicted and ground truth regions and is especially useful for evaluating imbalanced classes.

- Frames Per Second (FPS):
  This metric measures the inference speed of the model, i.e., how many frames the model can process per second during testing. It reflects the real-time applicability of the segmentation model.

5.1.2 Evaluation Results

Multi-Labels vs. Binary Labels: Table 5.1 and Figure 5.2 give inference comparisons of the two training strategies: Multi-Labels vs. Binary Labels (corresponding to introduction in Section 4.1.3). The results demonstrate that the model trained with binary labels consistently outperforms the that trained with multi-labels across all evaluation metrics on both the Face Synthetics and CelebMask-HQ datasets.

The visualization results further illustrate that both training strategies yield models with strong predictive performance on skin regions. Notably, the model trained with binary

Table 5.1.: The inference results of the two training strategies on two datasets.

	Face Synthetics[2]			CelebAMask-HQ[99]		
Model	**Pixel Accuracy**	**IoU score(Skin)**	**Dice score(Skin)**	**Pixel Accuracy**	**IoU score(Skin)**	**Dice score(Skin)**
UNet(multi labels)	0.96	0.86±0.06	0.94±0.04	0.80	0.81±0.05	0.93±0.05
UNet(binary labels)	**0.99**	**0.94±0.04**	**0.97±0.02**	**0.97**	**0.90±0.06**	**0.94±0.04**

Results are reported as mean ± standard deviation, with the best metric scores indicated in bold.

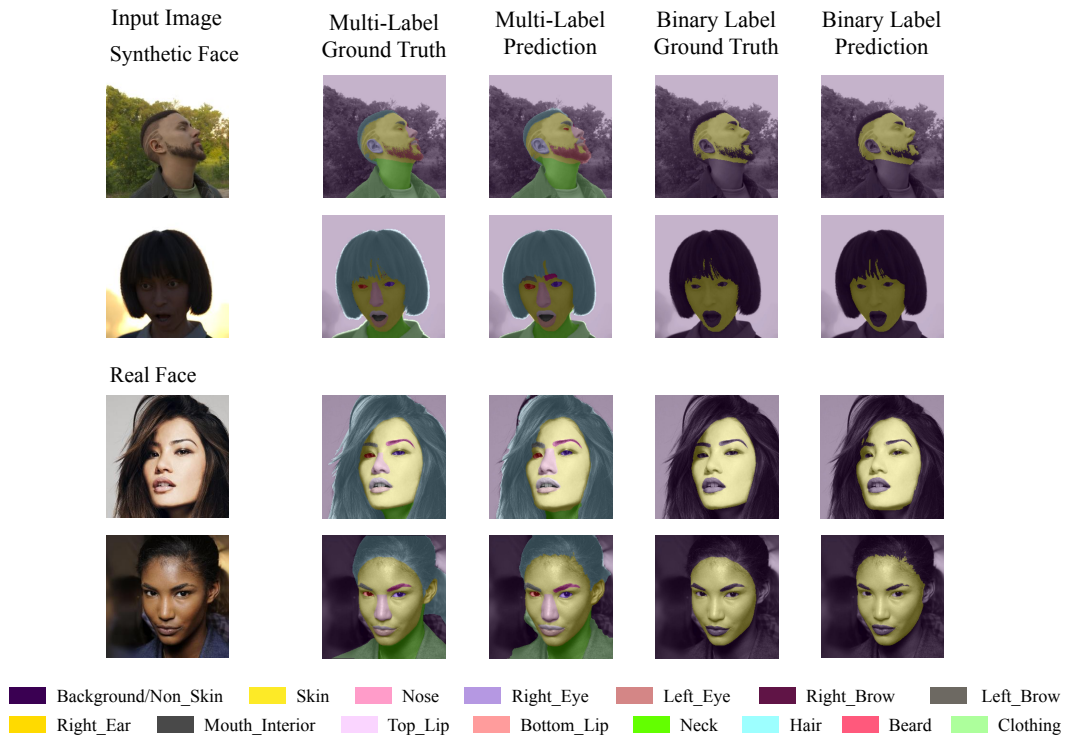


Figure 5.2.: Visual comparison of models by 2 different training strategies. Giving inference samples under 2 training strategies, each class are mapped with different color.

labels exhibits greater robustness during inference and achieves superior performance in certain fine-grained details. This confirms that binary segmentation is more effective for precise facial region extraction in PPGI applications.

Evaluation per Models:

Table 5.2.: Performance metrics of each model on the synthetic datasets.

Model	FPS*	Pixel Accuracy	IoU score(Skin)	Dice score(Skin)
UNet	**57.17**	**0.99**	**0.94±0.04**	**0.97±0.02**
UNet++	49.14	**0.99**	**0.94±0.04**	**0.97±0.02**
UNet3+	42.85	0.96	0.84±0.07	0.91±0.05
Swin-Unet	56.29	0.98	**0.94±0.04**	**0.97±0.02**
Mask2Former	19.97	0.97	0.93±0.05	0.96±0.03
SegNeXt	21.11	0.96	0.93±0.04	0.97±0.03

*FPS: frames/images per second during inference. Results are reported as mean ± standard deviation, with the best metric scores indicated in bold.

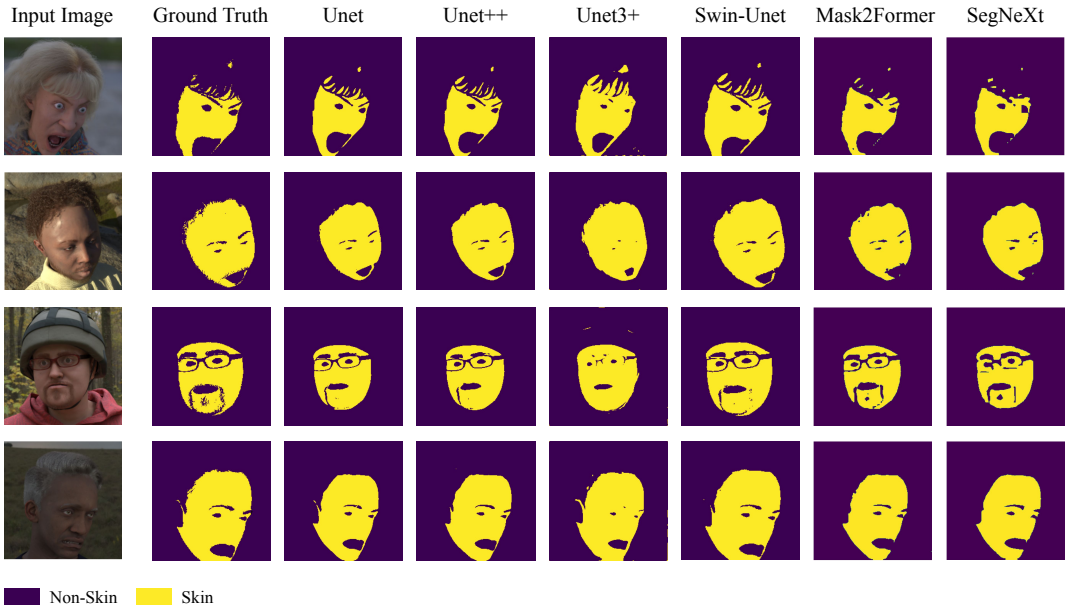


Figure 5.3.: Visualization of input images, the reference ground truth, and prediction results on synthetic test data from different models.

Table 5.2 presents the performance metrics of each model. As shown in the metrics table, U-Net achieves the highest inference speed while maintaining strong segmentation

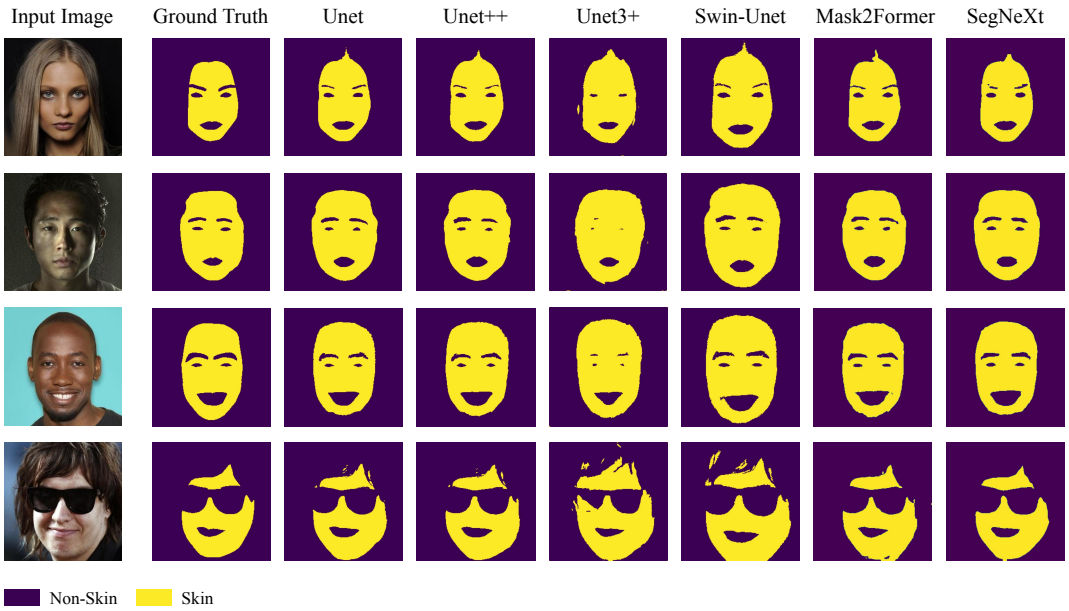


Figure 5.4.: Visualization of input images, the reference ground truth, and prediction results on real-face data from different models.

performance. Given its superior trade-off between real-time capability and segmentation accuracy, U-Net is selected as the optimal model for the downstream PPGI application.

Figure 5.4 and Figure 5.3 present predicted samples of each model on synthetic and real-face test data, respectively. The visualizations of these predictions align well with the results presented in Table 5.2. All models successfully identify skin regions, with U-Net and U-Net++ tending to better preserve facial structure and achieve more accurate boundaries in areas such as the eyes, mouth, and hairline. Meanwhile, transformer-based methods (Swin-Unet, Mask2Former, and SegNeXt) exhibit superior detail preservation, particularly in challenging regions like facial hair. It is also noteworthy that, as observed in the figure, certain regions (such as exposed scalp areas in the hairline or parting) are annotated as skin in the synthetic dataset, whereas they are labeled as hair (i.e., non-skin regions) in the real face dataset. This annotation inconsistency is one of the factors that contribute to the observed performance drop when evaluating on real data.

Edge Cases Analysis:

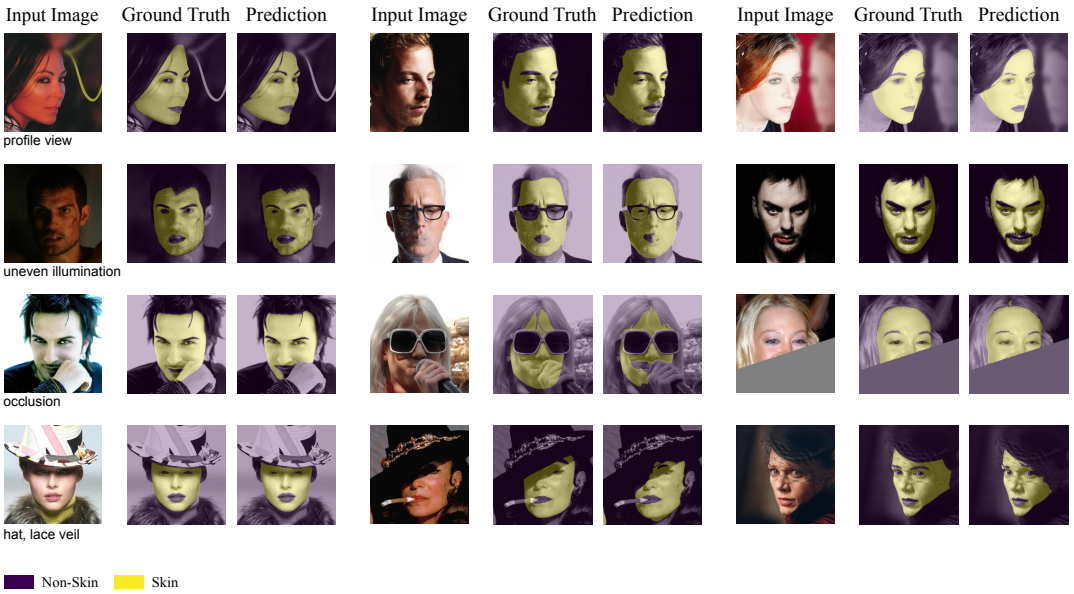


Figure 5.5.: Visualization of the prediction for several samples under edge cases. The model used is a U-Net (EfficientNet-B0 as backbone) trained with binary labels.

Figure 5.5 provides a visual comparison of facial skin segmentation results under challenging conditions such as illumination variation and occlusion. It can be observed that the segmentation model trained on synthetic data demonstrates strong robustness against uneven illumination, including side-lit facial appearances.

In cases of occlusion (e.g., by hands, glasses, or partially missing facial regions), the model still predicts the overall skin contour accurately, although minor errors are present. For subjects wearing hats, small inaccuracies are observed in areas under deep shadows cast by the brim.

As previously discussed, some performance drop is attributed to annotation inconsistencies between the synthetic training data and the real evaluation dataset. In the real data, occluded facial regions are often still labeled as facial skin, while the synthetic data lacks such ambiguity in annotation.

Additionally, substantial errors are observed when the face is partially occluded by lace veils. This is likely due to the absence of such elements in the synthetic training data, resulting in limited model generalization for semi-transparent occlusions.

5.2 PPGI Evaluation

This section introduces the evaluation in the branch of PPGI in this study. It is organized into two parts. The Evaluation Setup subsection outlines the experimental design, datasets, and metrics used to evaluate the pipeline. The Evaluation Results subsection presents and analyzes the performance across different tasks and subject conditions, highlighting the strengths and limitations of the approach—particularly the impact of skin segmentation quality on signal fidelity.

5.2.1 Evaluation Setup

Following the PPG processing branch described in Section 3.1, this stage involves computing heart rate (HR) from PPG signals extracted by the skin segmentation model across different datasets, and comparing the estimates to corresponding ground truth values. To ensure a fair comparison, both the predicted and reference HR values are computed using identical methodologies.

Two heart rate estimation methods are employed to evaluate the PPGI signals:

1. a pure frequency-domain approach based on the Fast Fourier Transform (FFT), and
2. a combined frequency- and time-domain approach based on peak detection guided by prior FFT estimation.

In the FFT-based method, the PPGI signal is bandpass filtered between 0.6–3.3 Hz to isolate the physiological frequency range. The dominant frequency component is then used to estimate the heart rate. This approach is straightforward and computationally efficient, but it may be sensitive to noise and spectral leakage.

To improve robustness, the peak detection method incorporates prior information from the FFT estimation. First, an initial heart rate guess ($\mathrm{HR}_{\mathrm{guess}}$) is computed as the average HR obtained from the FFT-based results. The PPGI signal is then bandpass filtered around this estimated heart rate with a passband of $[0.8 \times \mathrm{HR}_{\mathrm{guess}},\ 1.2 \times \mathrm{HR}_{\mathrm{guess}}]$, constrained within [0.4–4.0 Hz] to ensure physiological plausibility. Simple time-domain peak detection is subsequently applied to estimate the inter-beat intervals and compute HR.

This FFT-informed filtering step helps suppress irrelevant frequency components and enhances signal periodicity, thereby making the peak detection more accurate and robust, especially under noise, motion, or illumination variation.

For the UBFC dataset, HR is computed over 20-second sliding windows with a 10-second overlap; for the PURE dataset, 30-second windows with a 15-second overlap are used.

Finally, the estimated HR is evaluated using three metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Signal-to-Noise Ratio (SNR):

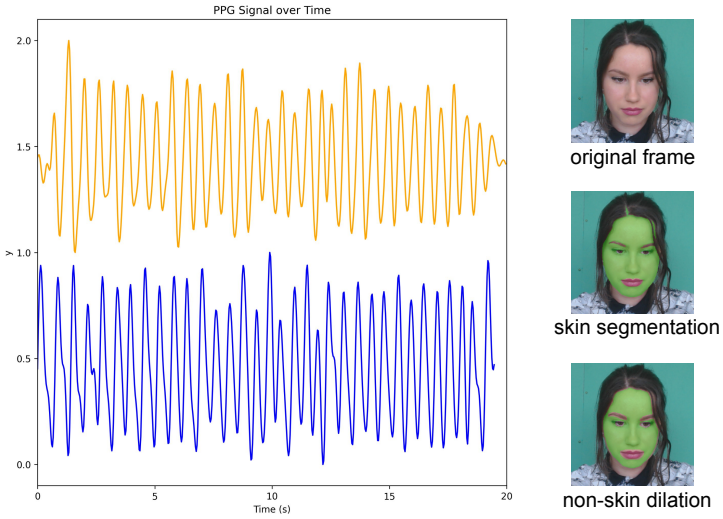
- **MAE** measures the average magnitude of prediction errors, offering a direct interpretation of how far the predictions deviate from the ground truth.
- **RMSE** emphasizes larger errors, making it suitable when significant deviations should be penalized more heavily.
- **SNR** reflects the clarity of the physiological signal by comparing signal power to noise power, thereby evaluating the quality and stability of the extracted waveform.

Together, these metrics provide a comprehensive assessment of the segmentation model's impact on downstream physiological signal estimation, both in terms of accuracy and signal fidelity.

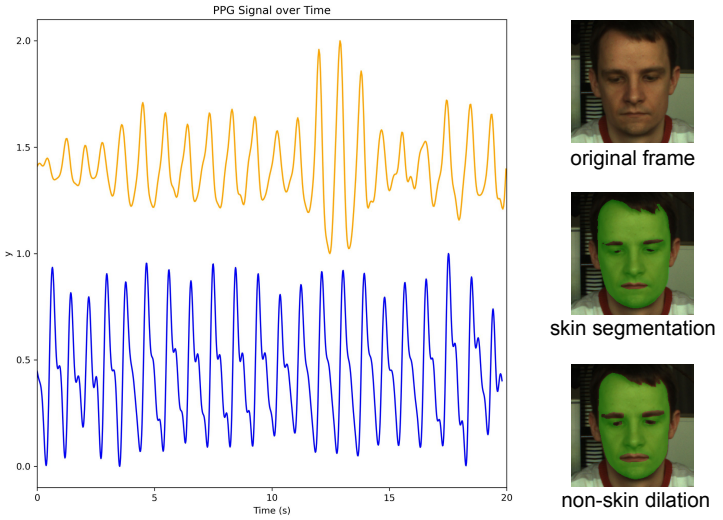
5.2.2 Evaluation Results

Figure 5.6 presents two sample waveform plots of extracted BVP signals with the ground truth PPG signals, together with visualizations of the corresponding ROIs selection.

Two representative samples are shown from the UBFC and PURE datasets, respectively. As shown on the left side, it illustrates the temporal alignment between the predicted PPG signal (orange) and the ground truth signal (blue) for two example participants. For improved readability, the first 20 seconds of the signals were normalized and shifted vertically by an offset of 1. As shown, the predicted signals largely follow the trend of the ground truth, capturing the periodic peaks corresponding to heartbeats, indicating that the skin segmentation-based approach provides reliable signal extraction across different subjects and datasets. The right side shows that we applied the dilation operation to the predicted non-skin regions



(a) UBFC - Participant 01 - HR: 102.75 BPM.



(b) PURE - Participant 01 - HR: 66.30 BPM.

Figure 5.6.: Visualization example results for PPG extraction and ROIs extraction. On the left side: result wave of extracted BVP (orange line) and ground truth (blue line) for each sample. On the right side (top to bottom): original frame, normal predicted ROIs, predicted ROIs with non-skin dilation option.

to decrease noise, especially in hairy skin regions (e.g., beard and eyebrows), which ensures an accurate ROIs extraction.

The Table 5.3 below shows the measured metrics MAE and RMSE of different methods on the datasets UBFC and PURE. These results indicate that the PPG signals extracted from our model-predicted ROIs exhibit a good correspondence with the ground truth signals.

Table 5.3.: Comparison of HR-estimation perfoemance on UBFC and PURE dataset.

Dataset	**Method**	**MAE** ↓	**RMSE** ↓	**SNR (dB)** ↑
UBFC[97]	Chen *et al.* [15]	1.05	1.63	-
	Sun *et al.* [100]	0.24	0.65	-
	Zhu *et al.* [101]	1.05	2.51	-
	Ours (FFT)	0.73±0.18	1.38±0.92	2.97±0.39
	Ours (FFT + Peak)	0.35±0.09	0.70±0.52	6.63±0.47
PURE[98]	Chen *et al.* [15])	3.86	12.08	-
	Sun *et al.* [100]	0.63	1.30	-
	Zhu *et al.* [101]	3.74	11.95	-
	Ours (FFT)	10.24±2.15	19.44±10.00	3.18±0.76
	Ours (FFT + Peak)	0.62±0.04	0.71±0.32	11.00±0.47

Table 5.4.: The results of UBFC-rPPG. [102]

	Method	**MAE** ↓	**RMSE** ↓
Deep learning method	Deepphys [51]	3.71	5.27
	rPPGNet [103]	3.24	4.97
	Physnet [104]	2.33	3.04
Traditional method	CHROM [13]	9.13	15.00
	GREEN [9]	20.90	29.33
	ICA [105]	8.62	13.54
	POS [10]	15.39	27.22

Quantitative evaluation using MAE and RMSE confirms that when compared with several recent rPPG methods, our approach achieves performance comparable to state-of-the-art

(SOTA) results on the same test dataset. Furthermore, higher SNR values, particularly on the PURE dataset, indicate that the estimation by the hybrid strategy yields cleaner and more noise-resilient signals, and the higher accuracy of hybrid estimation (FFT + peak detection)compared to pure FFT.

Table 5.4 summarizes the experimental results reported by Yang *et al.* [102]—performance of traditional methods and deep learning-based methods. In their approach, the UBFC-rPPG dataset was utilized for both training and testing deep learning-based methods, with 42 videos randomly split into a training set (37 videos) and a test set (5 videos). For a fair comparison with these methods, they evaluated traditional rPPG techniques solely on the same test set. In contrast, our proposed method is entirely trained on synthetic data, without incorporating any real-world samples during the training phase. The strong performance achieved on the UBFC test set further demonstrates the feasibility and effectiveness of using synthetic data for PPGI tasks.

In addition, we evaluate our model on samples recorded under the challenging and adverse scenarios introduced in the KISMED dataset (described in Section 4.2.1) to assess the robustness of our model for physiological signal extraction in difficult cases. We analyzed the data of the first subject (a 28-year-old male with fair skin) across 12 different test conditions. For PPG signal extraction, we exclusively use the ROS algorithm. For heart rate (HR) estimation, a fixed peak detection method is consistently applied. The results are shown in Figure 5.7 and Figure 5.8.

It can be observed that under ideal conditions (e.g., tasks v01 and v08), where subjects remain still and lighting is stable, both MAE and RMSE remain below 1 bpm, demonstrating the high accuracy and reliability of the POS method in controlled environments. As shown in the combined bar-line chart, these tasks also exhibit high mean IoU between frames ($\geq$0.83) and low IoU variance, indicating consistent segmentation across time. Notably, in task v06, the subject's movement away from the camera significantly reduces the facial region's proportion within the frame, leading to segmentation failure or inaccurate segmentation. As a result, the extracted ROI becomes invalid for physiological signal analysis.

In contrast, low-light conditions (e.g., v02, with ceiling light turned off) lead to a dramatic performance drop (MAE $>$ 26 bpm), with both skin area ratio and IoU metrics dropping, suggesting that insufficient illumination degrades segmentation quality and weakens chrominance-based signal extraction. Similarly, dynamic lighting (v03, ceiling light flickering) disrupts the stability of the extracted signal (MAE $>$ 35 bpm), as fluctuating skin tone and reflections interfere with spatial-temporal chrominance patterns critical to POS.

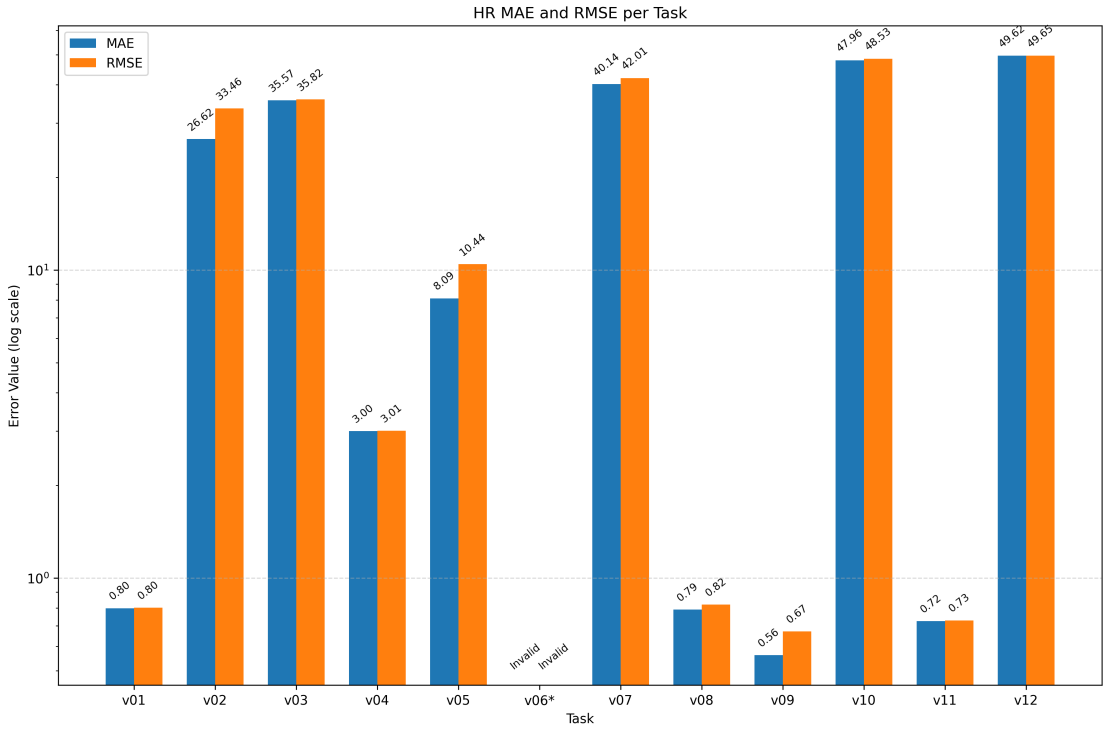


Figure 5.7.: HR MAE and RMSE of subject p001 for each task. The Y-axis is shown on a logarithmic scale for clearer comparative visualization.

In motion-heavy scenarios such as v07 (squats), v10 (in-place motion), and v12 (head turns and movement), the POS algorithm suffers from the highest errors (MAE $>$ 40 bpm). These tasks are characterized by low mean IoUs (0.80 – 0.86) and higher IoU variances, reflecting segmentation inconsistency due to rapid movement, occlusion, or face orientation changes. These issues disrupt both ROI tracking and the temporal smoothness of skin color variation, which are essential to POS.

Tasks like v05 (side lighting only) show moderate degradation in performance, aligning with reduced mean IoU and uneven skin exposure. This indicates that non-uniform illumination leads to spatial inconsistency in segmentation, affecting the reliability of the extracted rPPG signals.

On the other hand, partial facial occlusions and mild movements in v08, v09, and v11 cause only marginal increases in error (MAE $<$ 1 bpm), and segmentation metrics remain

high and stable. This observation underscores the robustness of the skin segmentation model, which helps maintain ROIs consistency and suppresses irrelevant background interference, thereby preserving signal quality even under imperfect conditions.

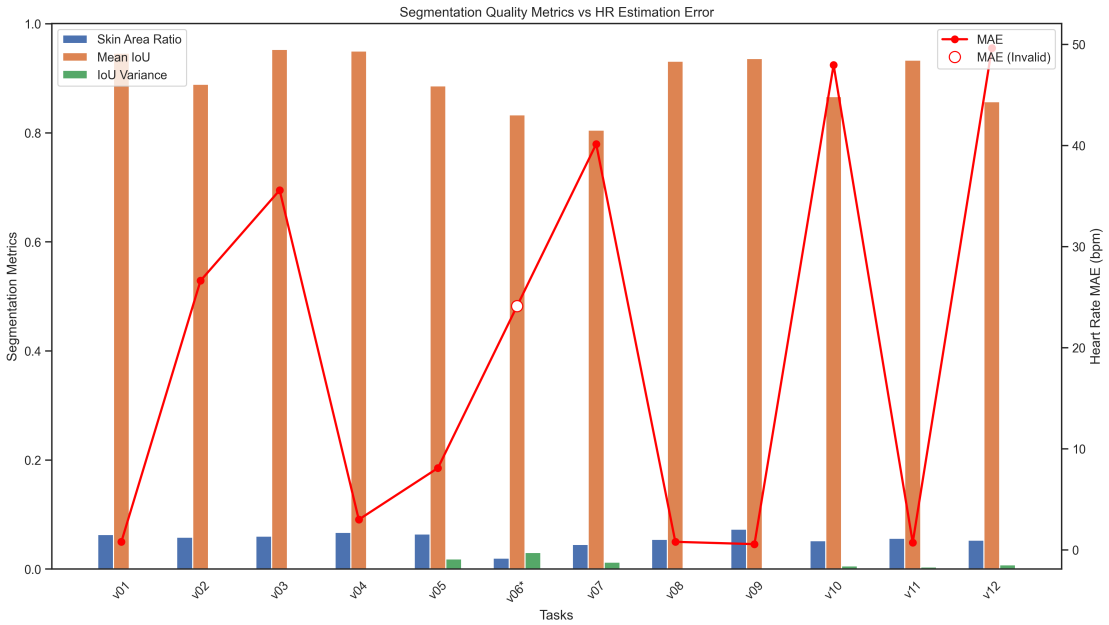


Figure 5.8.: Segmentation analysis for each task in subject p001 (inter-frame IoU vs. HR estimation error (MAE)). The skin area ratio refers to the predicted and post-processed skin region proportion relative to the entire face-included inference region (256×256 pixels).

Overall, the quantitative segmentation metrics (mean skin area ratio, mean frame-wise IoU, and IoU variance) correlate strongly with POS performance trends, as visualized in the combined plot. These results demonstrate that the skin segmentation model is not the limiting factor in poor-performing scenarios. Instead, challenges such as extreme lighting or subject motion dominate the error sources. The segmentation model contributes positively by stabilizing the ROI and filtering out background noise, thereby supporting signal quality under imperfect conditions.

Figure 5.9 illustrates representative sampled frames and the corresponding ROI detections across various scenarios, which aligns well with the preceding analysis. This further demonstrates, to some extent, that our trained skin segmentation model exhibits both

efficiency and robustness, provided that the facial region occupies a sufficiently large portion of the image.

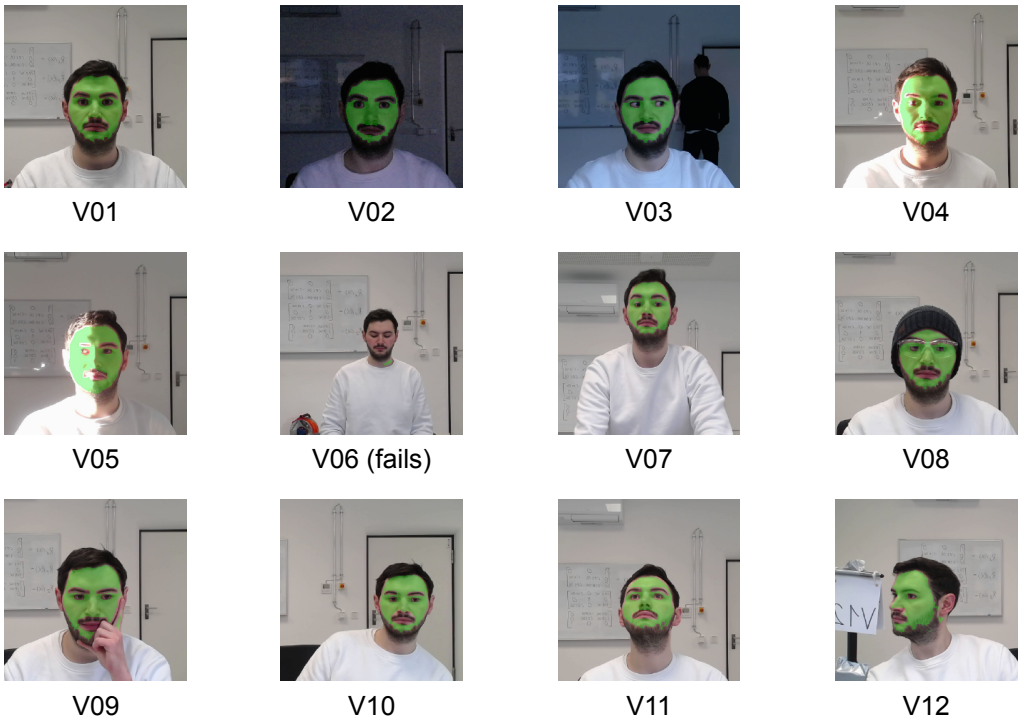


Figure 5.9.: ROI extraction samples for each task in subject p001.

6. Conclusion

This chapter presents the concluding remarks of this study on the implementation of a skin segmentation model based on synthetic face images using deep learning. It summarizes the key findings and contributions of the work, reflects on the limitations and challenges encountered during the research, and outlines possible directions for future improvement and extension. The chapter is organized into three sections: Summary, where the main outcomes and contributions are recapped; Discussion, which offers a critical analysis of the results and their implications; and Future Work, which proposes avenues for further development and research.

6.1 Summary

This study explored the development, evaluation, and application of a deep learning-based skin segmentation model trained on synthetic face images. The motivation for this research stems from the importance of robust and precise skin segmentation in various computer vision and biomedical applications, particularly in the context of remote physiological monitoring. Due to the difficulty of acquiring and annotating large-scale real facial datasets with dense pixel-level segmentation labels, the study adopted the Face Synthetics dataset as the primary training resource. This dataset, generated using advanced 3D rendering techniques, offers a rich variety of facial poses, lighting conditions, and skin tones, while maintaining consistent annotation quality.

The core model developed in this study is based on the U-Net architecture with an EfficientNetB0 backbone, which was selected due to its balance between accuracy and computational efficiency. Two labeling strategies were examined: a multi-class setting involving detailed facial part labels, and a binary setting focusing exclusively on skin versus non-skin regions. Experimental results demonstrated that the binary setting significantly outperformed the multi-class configuration for the purpose of skin segmentation, achieving

a pixel accuracy of 0.99, an Intersection-over-Union (IoU) score of 0.94, and a Dice score of 0.97 on the Face Synthetics test set.

The model's generalizability was evaluated on the CelebAMask-HQ dataset, where it also achieved satisfactory performance (IoU: 0.90, Dice: 0.94), although a slight drop in accuracy was observed due to domain differences between synthetic and real images. To benchmark the performance of the proposed architecture, several recent segmentation models were implemented and compared, including UNet++, UNet3+, Swin-Unet, SegNext, and Mask2Former. Among these, the binary U-Net model maintained top performance in terms of skin segmentation accuracy, highlighting its suitability as a reliable baseline.

The practical utility of the segmentation model was validated through its integration into a remote PPGI pipeline. Skin masks produced by the segmentation network were used to extract BVP signals from facial videos captured in publicly available datasets such as UBFC-RPPG, PURE, and KISMED. Notably, the PPGI pipeline remained effective under moderate facial motion, lighting variations, and partial occlusion, suggesting that the segmented skin regions provided stable and physiologically informative ROIs. These findings collectively demonstrate the feasibility and robustness of using synthetic data to train deep learning models for real-world skin segmentation tasks.

6.2 Discussion

The results presented in this thesis underscore several key insights regarding the design and application of deep learning-based skin segmentation systems. First and foremost, the use of synthetic data proved to be highly effective for training semantic segmentation models. The Face Synthetics dataset enabled the learning of fine-grained skin features across diverse facial appearances and environmental contexts, while avoiding the labor-intensive and error-prone process of manually labeling real data. The high accuracy achieved on both synthetic and real datasets confirms that models trained on synthetic data can generalize well when carefully designed.

However, the experiments also revealed the presence of a performance gap between synthetic and real-world datasets. While the model achieved excellent results on the synthetic validation set, a noticeable drop in performance was observed on CelebAMask-HQ, which suggests that domain shift remains a significant challenge. This drop can be attributed to differences in texture fidelity, lighting complexity, and noise patterns not captured in synthetic data. Addressing this issue will be crucial for deploying such models

in production environments, particularly in healthcare or surveillance contexts where reliability across demographics and environments is essential.

Nonetheless, a deeper examination of the results reveals several notable limitations that constrain the broader applicability and generalization of the proposed models. One of the primary constraints arises from the use of a fixed image resolution during training. All models were trained exclusively using images resized to 256×256 pixels (224×224 pixels for Swin-Transformer), without the inclusion of multi-scale data augmentation. While this design choice simplified training and ensured computational efficiency, it also limited the model's ability to adapt to varying image sizes and face scales during inference. In scenarios where the face occupies only a small portion of the image, such as in surveillance footage or wide-angle scenes, the model is more susceptible to background interference, which can significantly degrade prediction quality. The absence of scale diversity in the training phase hampers the model's robustness to variations in spatial resolution, particularly when facial features become less prominent.

Another limitation stems from the lack of training data containing transparent or semi-transparent occlusions, such as facial veils or mesh coverings. The Face Synthetics dataset, while diverse in facial appearances, hairstyles, lighting, and pose variations, does not include instances of faces partially obscured by translucent materials. As a result, the model is not exposed to such visual patterns during training and is likely to misclassify or completely overlook facial skin regions when they are covered by transparent masks or veils. This can be problematic in culturally or medically relevant contexts, where facial coverings are common. The inability to handle such occlusions reduces the practical reliability of the segmentation model and highlights the need for more inclusive and representative training data that reflects real-world variability.

The comparative analysis of different architectures offered valuable insights into the trade-offs between model complexity and segmentation accuracy. While transformer-based models such as Swin-Unet and Mask2Former are theoretically more powerful and have shown success in general-purpose segmentation tasks, they did not substantially outperform the simpler U-Net architecture in this specific task. In fact, U-Net with an EfficientNetB0 encoder provided the best results in the binary skin segmentation setting. This outcome suggests that for narrowly focused tasks such as skin segmentation, the additional complexity introduced by advanced architectures may not translate into proportional gains.

In the downstream PPGI application, the segmentation model played a crucial role in stabilizing the extraction of physiological signals. The POS algorithm used for BVP signal extraction benefited from the consistent and accurate skin masks, which helped to isolate

informative facial regions while excluding background and occluded areas. Analysis on the KISMED dataset demonstrated that the segmentation-enhanced PPGI system maintained low heart rate estimation errors even in challenging scenarios involving head movement or nonuniform illumination. These findings highlight the potential of segmentation-driven preprocessing in enhancing the robustness of non-contact physiological measurement systems.

In summary, while the proposed model achieves strong performance in several evaluation settings, its limitations—particularly regarding scale sensitivity and occlusion handling—point to clear directions for improvement. Addressing these issues will be essential for developing segmentation models that are not only accurate but also robust and deployable across diverse real-world applications.

6.3 Future Work

While the outcomes of this study are promising, several avenues remain open for future research and development. A key priority is to address the domain shift between synthetic and real data. Future work can investigate domain adaptation strategies such as adversarial domain adaptation, unsupervised feature alignment, or synthetic-to-real image translation using generative adversarial networks (GANs). These techniques may help improve model generalization by aligning feature distributions across domains.

One key limitation identified in the current implementation is the use of a fixed input image resolution throughout the training and evaluation stages. While this simplifies the model design and ensures computational efficiency, it inadvertently restricts the model's adaptability to variations in image scale during deployment. Future work should explore the integration of multi-scale training strategies, such as image pyramid augmentation, random resizing, and cropping at multiple resolutions. These techniques have been shown to enhance scale invariance and may significantly improve model robustness when handling images in which facial regions appear at varying sizes, positions, or distances from the camera. By training the model to recognize facial skin features across a broader range of spatial contexts, it becomes better equipped to generalize to real-world scenarios, including wide-angle shots and partially zoomed-out faces.

In conjunction with scale augmentation, another important direction is the dynamic adjustment of regions of interest (ROIs) based on face detection confidence or saliency estimation. When a face occupies only a small portion of the image, background elements—such as hair, clothing, or environmental textures—can dominate the receptive field and reduce

segmentation accuracy. Incorporating a pre-segmentation face localization step or an adaptive cropping mechanism could allow the model to focus more precisely on relevant facial regions, thereby minimizing background noise and improving the reliability of predictions.

Another notable limitation of the current dataset and model is the absence of training samples involving facial occlusions with transparent or semi-transparent materials, such as veils, medical masks, or costume accessories. Such occlusions are common in real-world scenarios across various cultural, social, and occupational contexts. Future work should address this shortcoming by augmenting the training data with realistic samples that feature transparent or partially transparent facial coverings. This could be achieved either by rendering additional synthetic data that includes such elements or by utilizing generative data augmentation techniques to overlay occlusions on existing images in a physically plausible manner. By exposing the model to these more challenging visual conditions during training, it may learn to better infer skin regions even when partial obstruction occurs, thereby improving prediction robustness in unconstrained environments.

In terms of model architecture, while this study demonstrated that U-Net and its variants perform remarkably well in the task of binary skin segmentation, future work may explore more advanced backbones or hybrid encoder designs. For instance, transformer-CNN fusion models or self-attention-enhanced UNet variants could potentially combine local detail sensitivity with improved global context awareness. Moreover, leveraging pretraining on large-scale human parsing or facial analysis datasets may further enhance performance, especially when fine-tuning on real-world data.

Moreover, integrating the segmentation and physiological signal estimation stages into a single end-to-end trainable framework may offer significant benefits. Instead of treating segmentation as a preprocessing step, future models can jointly learn segmentation and signal extraction objectives, allowing for direct optimization of physiological measurement accuracy. Such an approach may lead to improved temporal stability and reduced error propagation.

Lastly, real-time deployment remains a practical challenge and an important goal. Techniques such as model quantization, pruning, and hardware-aware neural architecture search (NAS) could be employed to reduce model size and latency. Real-time skin segmentation and PPGI pipelines could have immediate applications in telemedicine, fitness monitoring, and emotion recognition, making this line of research both impactful and timely.

Bibliography

[1] Kunyoung Lee, Jaemu Oh, Hojoon You, et al. “Improving Remote Photoplethysmography Performance through Deep-Learning-Based Real-Time Skin Segmentation Network”. In: *Electronics* 12.17 (2023). ISSN: 2079-9292. DOI: 10.3390/electronics12173729. URL: https://www.mdpi.com/2079-9292/12/17/3729.

[2] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, et al. *Fake It Till You Make It: Face analysis in the wild using synthetic data alone*. 2021. arXiv: 2109.15102 [cs.CV]. URL: https://arxiv.org/abs/2109.15102.

[3] Matthieu Scherpf, Hannes Ernst, Hagen Malberg, et al. “DeepPerfusion: A Comprehensible Two-Branched Deep Learning Architecture for High-Precision Blood Volume Pulse Extraction Based on Imaging Photoplethysmography”. In: (Aug. 2024). DOI: 10.36227/techrxiv.172503790.08194939/v1. URL: http://dx.doi.org/10.36227/techrxiv.172503790.08194939/v1.

[4] A. B. Hertzman. “Photoelectric Plethysmography of the Fingers and Toes in Man”. In: *Experimental Biology and Medicine* 37.3 (1937), pp. 529–534. DOI: 10.3181/00379727-37-9630.

[5] John Allen. “Photoplethysmography and its application in clinical physiological measurement”. In: *Physiological measurement* 28.3 (2007), R1.

[6] Lionel Tarassenko, Mauricio Villarroel, Andrea Guazzi, et al. “Non-contact video-based vital sign monitoring using ambient light and auto-regressive models”. In: *Physiological Measurement* 35.5 (2014), pp. 807–831. DOI: 10.1088/0967-3334/35/5/807.

[7] Yu Sun, Sijung Hu, Vicente Azorin-Peris, et al. “Motion-compensated noncontact imaging photoplethysmography to monitor cardiorespiratory status during exercise”. In: *Journal of biomedical optics* 16.7 (2011), pp. 077010–077010.

[8] T Tamura, Y Maeda, M Sekine, et al. “Wearable photoplethysmographic sensors—past and present”. In: *Electronics* 3.2 (2014), pp. 282–302.

[9] Wim Verkruysse, Lars O Svaasand, and J Stuart Nelson. “Remote plethysmographic imaging using ambient light.” In: *Optics express* 16.26 (2008), pp. 21434–21445.

[10] Wenjin Wang, Albertus C. den Brinker, Sander Stuijk, et al. “Algorithmic Principles of Remote PPG”. In: *IEEE Transactions on Biomedical Engineering* 64.7 (2017), pp. 1479–1491. DOI: 10.1109/TBME.2016.2609282.

[11] Ming-Zher Poh, Daniel J. McDuff, and Rosalind W. Picard. “Non-contact, automated cardiac pulse measurements using video imaging and blind source separation.” In: *Opt. Express* 18.10 (May 2010), pp. 10762–10774. DOI: 10.1364/OE.18.010762. URL: https://opg.optica.org/oe/abstract.cfm?URI=oe-18-10-10762.

[12] Timon Blöcher, Johannes Schneider, Markus Schinle, et al. “An online PPGI approach for camera based heart rate monitoring using beat-to-beat detection”. In: *2017 IEEE Sensors Applications Symposium (SAS)*. 2017, pp. 1–6. DOI: 10.1109/SAS.2017.7894052.

[13] Gerard de Haan and Vincent Jeanne. “Robust Pulse Rate From Chrominance-Based rPPG”. In: *IEEE Transactions on Biomedical Engineering* 60.10 (2013), pp. 2878–2886. DOI: 10.1109/TBME.2013.2266196.

[14] Jordan Fine, Kelly L. Branan, Alan J. Rodriguez, et al. “Sources of Inaccuracy in Photoplethysmography for Continuous Cardiovascular Monitoring”. In: *Biosensors (Basel)* 11.4 (Apr. 2021), p. 126. DOI: 10.3390/bios11040126. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8073123/.

[15] Shutao Chen, Kwan-Long Wong, Jing-Wei Chin, et al. “DiffPhys: Enhancing Signal-to-Noise Ratio in Remote Photoplethysmography Signal Using a Diffusion Model Approach”. In: *Bioengineering* 11.8 (2024). ISSN: 2306-5354. DOI: 10.3390/bioengineering11080743. URL: https://www.mdpi.com/2306-5354/11/8/743.

[16] Xiaobai Li, Jie Chen, Guoying Zhao, et al. “Remote Heart Rate Measurement from Face Videos under Realistic Situations”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 4264–4271. DOI: 10.1109/CVPR.2014.543.

[17] Wei Chen, Daniel McDuff, and Andreas Bulling. “Physiological Measurement via Domain Knowledge-Guided Deep Learning”. In: *IEEE Pervasive Computing* 17.3 (2018), pp. 28–38.

[18] Haoyu Qiu, Xiaoming Liu, Deyuan Sun, et al. “Skin-based Region Selection for Robust Remote Photoplethysmography”. In: *Biomedical Signal Processing and Control* 73 (2022), p. 103454. DOI: 10.1016/j.bspc.2021.103454.

[19] Daniel J McDuff, Justin R Estepp, Alyssa M Piasecki, et al. “A survey of remote optical photoplethysmographic imaging methods”. In: *2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE. 2015, pp. 6398–6404.

[20] Raymundo Cassani, Abhishek Tiwari, and Tiago H Falk. “Optimal filter characterization for photoplethysmography-based pulse rate and pulse power spectrum estimation”. In: *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE. 2020, pp. 914–917.

[21] Daniel McDuff, Sarah Gontarek, and Rosalind W Picard. “Improvements in remote cardiopulmonary measurement using a five band digital camera”. In: *IEEE Transactions on Biomedical Engineering* 61.10 (2014), pp. 2593–2601.

[22] Praveen Kakumanu, Sokratis Makrogiannis, and Nikolaos Bourbakis. “A survey of skin-color modeling and detection methods”. In: *Pattern recognition* 40.3 (2007), pp. 1106–1122.

[23] W. Zhao, R. Chellappa, P. J. Phillips, et al. “Face recognition: A literature survey”. In: 35.4 (Dec. 2003), pp. 399–458. ISSN: 0360-0300. DOI: 10.1145/954339.954342. URL: https://doi.org/10.1145/954339.954342.

[24] Amirhossein Dadashzadeh, Alireza Tavakoli Targhi, Maryam Tahmasbi, et al. “HGR-Net: a fusion network for hand gesture segmentation and recognition”. In: *IET Computer Vision* 13.8 (2019), pp. 700–707.

[25] Byoung Chul Ko. “A brief review of facial emotion recognition based on visual information”. In: *sensors* 18.2 (2018), p. 401.

[26] S. Kolkur, D. Kalbande, P. Shimpi, et al. “Human Skin Detection Using RGB, HSV and YCbCr Color Models”. In: *Proceedings of the International Conference on Communication and Signal Processing 2016 (ICCASP 2016)*. iccasp-16. Atlantis Press, 2017. DOI: 10.2991/iccasp-16.2017.51. URL: http://dx.doi.org/10.2991/iccasp-16.2017.51.

[27] S.L. Phung, A. Bouzerdoum, and D. Chai. “Skin segmentation using color and edge information”. In: *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings*. Vol. 1. 2003, 525–528 vol.1. DOI: 10.1109/ISSPA.2003.1224755.

[28] Michael J. Jones and James M. Rehg. “Statistical Color Models with Application to Skin Detection”. In: *International Journal of Computer Vision* 46.1 (2002), pp. 81–96. ISSN: 1573-1405. DOI: 10.1023/A:1013200319198. URL: https://doi.org/10.1023/A:1013200319198.

[29] J. Kovac, P. Peer, and F. Solina. “Human skin color clustering for face detection”. In: *The IEEE Region 8 EUROCON 2003. Computer as a Tool.* Vol. 2. 2003, 144–148 vol.2. DOI: 10.1109/EURCON.2003.1248169.

[30] Zhiwei Jiang, Min Yao, and Wei Jiang. “Skin Detection Using Color, Texture and Space Information”. In: *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)*. Vol. 3. 2007, pp. 366–370. DOI: 10.1109/FSKD.2007.518.

[31] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 3431–3440. DOI: 10.1109/CVPR.2015.7298965.

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. arXiv: 1505.04597 [cs.CV]. URL: https://arxiv.org/abs/1505.04597.

[33] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, et al. *UNet++: A Nested U-Net Architecture for Medical Image Segmentation*. 2018. arXiv: 1807.10165 [cs.CV]. URL: https://arxiv.org/abs/1807.10165.

[34] Huimin Huang, Lanfen Lin, Ruofeng Tong, et al. *UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation*. 2020. arXiv: 2004.08790 [eess.IV]. URL: https://arxiv.org/abs/2004.08790.

[35] Jieneng Chen, Yongyi Lu, Qihang Yu, et al. *TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation*. 2021. arXiv: 2102.04306 [cs.CV]. URL: https://arxiv.org/abs/2102.04306.

[36] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV]. URL: https://arxiv.org/abs/2010.11929.

[37] Hugo Touvron, Matthieu Cord, Matthijs Douze, et al. *Training data-efficient image transformers & distillation through attention*. 2021. arXiv: 2012.12877 [cs.CV]. URL: https://arxiv.org/abs/2012.12877.

[38] Ze Liu, Yutong Lin, Yue Cao, et al. “Swin transformer: Hierarchical vision transformer using shifted windows”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 10012–10022.

[39] Hu Cao, Yueyue Wang, Joy Chen, et al. *Swin-Unet: Unet-like Pure Transformer for Medical Image Segmentation*. 2021. arXiv: 2105.05537 [eess.IV]. URL: https://arxiv.org/abs/2105.05537.

[40] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, et al. *SegNeXt: Rethinking Convolutional Attention Design for Semantic Segmentation*. 2022. arXiv: 2209.08575 [cs.CV]. URL: https://arxiv.org/abs/2209.08575.

[41] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *Nature* 521.7553 (May 2015), pp. 436–444. DOI: 10.1038/nature14539.

[42] Peike Li, Yunqiu Xu, Yunchao Wei, et al. *Self-Correction for Human Parsing*. 2019. arXiv: 1910.09777 [cs.CV]. URL: https://arxiv.org/abs/1910.09777.

[43] Adam Kortylewski, Bernhard Egger, Andreas Schneider, et al. “Analyzing and Reducing the Damage of Dataset Bias to Face Recognition With Synthetic Data”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2019, pp. 2261–2268. DOI: 10.1109/CVPRW.2019.00279.

[44] Fadi Boutros, Vitomir Struc, Julian Fierrez, et al. “Synthetic data for face recognition: Current state and future prospects”. In: *Image and Vision Computing* 135 (2023), p. 104688. ISSN: 0262-8856. DOI: https://doi.org/10.1016/j.imavis.2023.104688. URL: https://www.sciencedirect.com/science/article/pii/S0262885623000628.

[45] Volker Blanz and Thomas Vetter. “A morphable model for the synthesis of 3D faces”. In: *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 2023, pp. 157–164.

[46] Bernhard Egger, William A. P. Smith, Ayush Tewari, et al. *3D Morphable Face Models – Past, Present and Future*. 2020. arXiv: 1909.01815 [cs.CV]. URL: https://arxiv.org/abs/1909.01815.

[47] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML]. URL: https://arxiv.org/abs/1406.2661.

[48] Haoyu Zhang, Marcel Grimmer, Raghavendra Ramachandra, et al. *On the Applicability of Synthetic Data for Face Recognition*. 2021. arXiv: 2104.02815 [cs.CV]. URL: https://arxiv.org/abs/2104.02815.

[49] Tero Karras, Samuli Laine, and Timo Aila. *A Style-Based Generator Architecture for Generative Adversarial Networks*. 2019. arXiv: 1812.04948 [cs.NE]. URL: https://arxiv.org/abs/1812.04948.

[50] Eli Friedman, Assaf Lehr, Alexey Gruzdev, et al. *Knowing the Distance: Understanding the Gap Between Synthetic and Real Data For Face Parsing*. 2023. arXiv: 2303.15219 [cs.CV]. URL: https://arxiv.org/abs/2303.15219.

[51] Weixuan Chen and Daniel McDuff. *DeepPhys: Video-Based Physiological Measurement Using Convolutional Attention Networks*. 2018. arXiv: 1805.07888 [cs.CV]. URL: https://arxiv.org/abs/1805.07888.

[52] Kai Li, Hannes Rüdiger, and Tjalf Ziemssen. "Spectral Analysis of Heart Rate Variability: Time Window Matters". In: *Frontiers in Neurology* 10 (May 2019), p. 545. DOI: 10.3389/fneur.2019.00545. URL: https://www.frontiersin.org/articles/10.3389/fneur.2019.00545/full.

[53] M.P. Tarvainen, P.O. Ranta-aho, and P.A. Karjalainen. "An advanced detrending method with application to HRV analysis". In: *IEEE Transactions on Biomedical Engineering* 49.2 (2002), pp. 172–175. DOI: 10.1109/10.979357.

[54] P. Welch. "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms". In: *IEEE Transactions on Audio and Electroacoustics* 15.2 (1967), pp. 70–73. DOI: 10.1109/TAU.1967.1161901.

[55] Yiru Sun and Nitish V Thakor. "Photoplethysmography Revisited: From Contact to Noncontact, From Point to Imaging". In: *IEEE Transactions on Biomedical Engineering* 63.3 (2016), pp. 463–477. DOI: 10.1109/TBME.2015.2476337. URL: https://doi.org/10.1109/TBME.2015.2476337.

[56] Maurice Rohr, Monika Hoog Antink, Sebastian Dill, et al. "Using Consumer Camera and Custom Firmware to Monitor Heart Rate in Terminally Ill Children during Music Therapy". In: *2023 Computing in Cardiology (CinC)*. Vol. 50. 2023, pp. 1–4.

[57] Zhilin Zhang. "Photoplethysmography-Based Heart Rate Monitoring in Physical Activities via Joint Sparse Spectrum Reconstruction". In: *IEEE Transactions on Biomedical Engineering* 62.8 (2015), pp. 1902–1910. DOI: 10.1109/TBME.2015.2406332.

[58] Harishchandra Dubey, Ramdas Kumaresan, and Kunal Mankodiya. *Harmonic Sum-based Method for Heart Rate Estimation using PPG Signals Affected with Motion Artifacts*. 2016. arXiv: 1610.05112 [cs.CY]. URL: https://arxiv.org/abs/1610.05112.

[59] Sergey Tulyakov, Xavier Alameda-Pineda, Elisa Ricci, et al. “Self-Adaptive Matrix Completion for Heart Rate Estimation from Face Videos under Realistic Conditions”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2396–2404. DOI: 10.1109/CVPR.2016.263.

[60] Georg Lempe, Sebastian Zaunseder, Tom Wirthgen, et al. “ROI selection for remote photoplethysmography”. In: *Bildverarbeitung für die Medizin 2013: Algorithmen-Systeme-Anwendungen. Proceedings des Workshops vom 3. bis 5. März 2013 in Heidelberg*. Springer. 2013, pp. 99–103.

[61] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, et al. “Pyramid scene parsing network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2881–2890.

[62] Liang-Chieh Chen, Yukun Zhu, George Papandreou, et al. *Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation*. 2018. arXiv: 1802.02611 [cs.CV]. URL: https://arxiv.org/abs/1802.02611.

[63] Changqian Yu, Jingbo Wang, Chao Peng, et al. *BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation*. 2018. arXiv: 1808.00897 [cs.CV]. URL: https://arxiv.org/abs/1808.00897.

[64] Changqian Yu, Changxin Gao, Jingbo Wang, et al. *BiSeNet V2: Bilateral Network with Guided Aggregation for Real-time Semantic Segmentation*. 2020. arXiv: 2004.02147 [cs.CV]. URL: https://arxiv.org/abs/2004.02147.

[65] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. *Per-Pixel Classification is Not All You Need for Semantic Segmentation*. 2021. arXiv: 2107.06278 [cs.CV]. URL: https://arxiv.org/abs/2107.06278.

[66] Bowen Cheng, Ishan Misra, Alexander G. Schwing, et al. *Masked-attention Mask Transformer for Universal Image Segmentation*. 2022. arXiv: 2112.01527 [cs.CV]. URL: https://arxiv.org/abs/2112.01527.

[67] Xiao Liu, Xiaofei Si, and Jiangtao Xie. *3rd Place Solution for Short-video Face Parsing Challenge*. 2021. arXiv: 2106.07409 [cs.CV]. URL: https://arxiv.org/abs/2106.07409.

[68] Ke Gong, Xiaodan Liang, Dongyu Zhang, et al. *Look into Person: Self-supervised Structure-sensitive Learning and A New Benchmark for Human Parsing*. 2017. arXiv: 1703.05446 [cs.CV]. URL: https://arxiv.org/abs/1703.05446.

[69] Jia Deng, R. Socher, Li Fei-Fei, et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*. Vol. 00. June 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848. URL: https://ieeexplore.ieee.org/abstract/document/5206848/.

[70] Bolei Zhou, Hang Zhao, Xavier Puig, et al. “Scene Parsing through ADE20K Dataset”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5122–5130.

[71] Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, et al. “Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?” In: *IEEE Transactions on Medical Imaging* 35.5 (2016), pp. 1299–1312. DOI: 10.1109/TMI.2016.2535302.

[72] German Ros, Laura Sellart, Joanna Materzynska, et al. “The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 3234–3243. DOI: 10.1109/CVPR.2016.352.

[73] Stephan R. Richter, Vibhav Vineet, Stefan Roth, et al. *Playing for Data: Ground Truth from Computer Games*. 2016. arXiv: 1608.02192 [cs.CV]. URL: https://arxiv.org/abs/1608.02192.

[74] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, et al. *Learning from Simulated and Unsupervised Images through Adversarial Training*. 2017. arXiv: 1612.07828 [cs.CV]. URL: https://arxiv.org/abs/1612.07828.

[75] Judy Hoffman, Eric Tzeng, Taesung Park, et al. *CyCADA: Cycle-Consistent Adversarial Domain Adaptation*. 2017. arXiv: 1711.03213 [cs.CV]. URL: https://arxiv.org/abs/1711.03213.

[76] Alexey Dosovitskiy, German Ros, Felipe Codevilla, et al. “CARLA: An open urban driving simulator”. In: *Conference on robot learning*. PMLR. 2017, pp. 1–16.

[77] Marius Cordts, Mohamed Omran, Sebastian Ramos, et al. *The Cityscapes Dataset for Semantic Urban Scene Understanding*. 2016. arXiv: 1604.01685 [cs.CV]. URL: https://arxiv.org/abs/1604.01685.

[78] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, et al. *ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation*. 2019. arXiv: 1811.12833 [cs.CV]. URL: https://arxiv.org/abs/1811.12833.

[79] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. “The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions”. In: *Scientific Data* 5.1 (Aug. 2018). ISSN: 2052-4463. DOI: 10.1038/sdata.2018.161. URL: http://dx.doi.org/10.1038/sdata.2018.161.

[80] Agisilaos Chartsias, Thomas Joyce, Rohan Dharmakumar, et al. “Adversarial image synthesis for unpaired multi-modal cardiac data”. In: *Simulation and Synthesis in Medical Imaging: Second International Workshop, SASHIMI 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 10, 2017, Proceedings 2*. Springer. 2017, pp. 3–13.

[81] Tianchu Guo, Youngsung Kim, Hui Zhang, et al. “Residual encoder decoder network and adaptive prior for face parsing”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32. 1. 2018.

[82] Zhen Wei, Si Liu, Yao Sun, et al. “Accurate Facial Image Parsing at Real-Time Speed”. In: *IEEE Transactions on Image Processing* 28.9 (2019), pp. 4659–4670. DOI: 10.1109/TIP.2019.2909652.

[83] Jinpeng Lin, Hao Yang, Dong Chen, et al. *Face Parsing with RoI Tanh-Warping*. 2019. arXiv: 1906.01342 [cs.CV]. URL: https://arxiv.org/abs/1906.01342.

[84] Yinglu Liu, Hailin Shi, Hao Shen, et al. “A new dataset and boundary-attention semantic segmentation for face parsing”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 07. 2020, pp. 11637–11644.

[85] Gusi Te, Yinglu Liu, Wei Hu, et al. *Edge-aware Graph Representation Learning and Reasoning for Face Parsing*. 2020. arXiv: 2007.11240 [cs.CV]. URL: https://arxiv.org/abs/2007.11240.

[86] Haibo Qiu, Baosheng Yu, Dihong Gong, et al. *SynFace: Face Recognition with Synthetic Data*. 2021. arXiv: 2108.07960 [cs.CV]. URL: https://arxiv.org/abs/2108.07960.

[87] S. Saxena, K. Lal, and S. Joshi. “Retinal Vessel Segmentation Using Blending-Based Conditional Generative Adversarial Networks”. In: *Computer Analysis of Images and Patterns (CAIP 2021)*. Ed. by Nicolas Tsapatsoulis, Aresti Panayides, Theocharis Theocharides, et al. Vol. 13052. Lecture Notes in Computer Science. Springer, Cham, 2021, pp. 135–146. ISBN: 978-3-030-89127-5. DOI: 10.1007/978-3-030-89128-2_13.

[88] Abdulrahman Kerim. *Synthetic Data for Machine Learning*. Chapter 19: The Domain Gap Problem in ML. Packt Publishing, 2023. URL: https://subscription.packtpub.com/book/data/9781803245409/19/ch19lvl1sec84/the-domain-gap-problem-in-ml.

[89] Nima Tajbakhsh, Laura Jeyaseelan, Qian Li, et al. *Embracing Imperfect Datasets: A Review of Deep Learning Solutions for Medical Image Segmentation*. 2020. arXiv: 1908.10454 [eess.IV]. URL: https://arxiv.org/abs/1908.10454.

[90] Mingxing Tan and Quoc V. Le. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. 2020. arXiv: 1905.11946 [cs.LG]. URL: https://arxiv.org/abs/1905.11946.

[91] Mingxing Tan, Bo Chen, Ruoming Pang, et al. *MnasNet: Platform-Aware Neural Architecture Search for Mobile*. 2019. arXiv: 1807.11626 [cs.CV]. URL: https://arxiv.org/abs/1807.11626.

[92] Jie Hu, Li Shen, Samuel Albanie, et al. *Squeeze-and-Excitation Networks*. 2019. arXiv: 1709.01507 [cs.CV]. URL: https://arxiv.org/abs/1709.01507.

[93] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, et al. *End-to-End Object Detection with Transformers*. 2020. arXiv: 2005.12872 [cs.CV]. URL: https://arxiv.org/abs/2005.12872.

[94] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV]. URL: https://arxiv.org/abs/1512.03385.

[95] Pavel Iakubovskii. *Segmentation Models Pytorch*. https://github.com/qubvel/segmentation_models.pytorch. Accessed: 2025-04-10. 2019.

[96] MMSegmentation Contributors. *OpenMMLab Semantic Segmentation Toolbox and Benchmark*. https://github.com/open-mmlab/mmsegmentation/tree/main/configs/segnext. Accessed: 2025-04-15. 2020.

[97] S. Bobbia, R. Macwan, Y. Benezeth, et al. "Unsupervised skin tissue segmentation for remote photoplethysmography". In: *Pattern Recognition Letters* 83 (2017), pp. 19–27. DOI: 10.1016/j.patrec.2016.09.007.

[98] R. Stricker, S. Müller, and H.-M. Gross. "Non-contact Video-based Pulse Rate Measurement on a Mobile Service Robot". In: *Proceedings of the 23rd IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. Edinburgh, Scotland, UK: IEEE, 2014, pp. 1056–1062.

[99] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, et al. “MaskGAN: Towards Diverse and Interactive Facial Image Manipulation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[100] Xiujuan Sun, Ying Su, Xiankai Hou, et al. “Research on Heart Rate Detection from Facial Videos Based on an Attention Mechanism 3D Convolutional Neural Network”. In: *Electronics* 14.2 (2025). ISSN: 2079-9292. DOI: 10.3390/electronics14020269. URL: https://www.mdpi.com/2079-9292/14/2/269.

[101] Dali Zhu, Wenli Zhang, Hualin Zeng, et al. *rFaceNet: An End-to-End Network for Enhanced Physiological Signal Extraction through Identity-Specific Facial Contours*. 2024. arXiv: 2403.09034 [cs.CV]. URL: https://arxiv.org/abs/2403.09034.

[102] Ze Yang, Haofei Wang, and Feng Lu. *Assessment of Deep Learning-based Heart Rate Estimation using Remote Photoplethysmography under Different Illuminations*. 2022. arXiv: 2107.13193 [cs.CV]. URL: https://arxiv.org/abs/2107.13193.

[103] Zitong Yu, Wei Peng, Xiaobai Li, et al. *Remote Heart Rate Measurement from Highly Compressed Facial Videos: an End-to-end Deep Learning Solution with Video Enhancement*. 2019. arXiv: 1907.11921 [eess.IV]. URL: https://arxiv.org/abs/1907.11921.

[104] Zitong Yu, Xiaobai Li, and Guoying Zhao. *Remote Photoplethysmograph Signal Measurement from Facial Videos Using Spatio-Temporal Networks*. 2019. arXiv: 1905.02419 [cs.CV]. URL: https://arxiv.org/abs/1905.02419.

[105] Ming-Zher Poh, Daniel J McDuff, and Rosalind W Picard. “Advancements in noncontact, multiparameter physiological measurements using a webcam”. In: *IEEE transactions on biomedical engineering* 58.1 (2010), pp. 7–11.

A. Appendix

To better understand the influence of label granularity on model performance, this appendix explores the analysis of label distributions in the Face Synthetics dataset, which also serves as a reference for determining class-wise loss weighting during the training process.

A.1 Labels Analysis

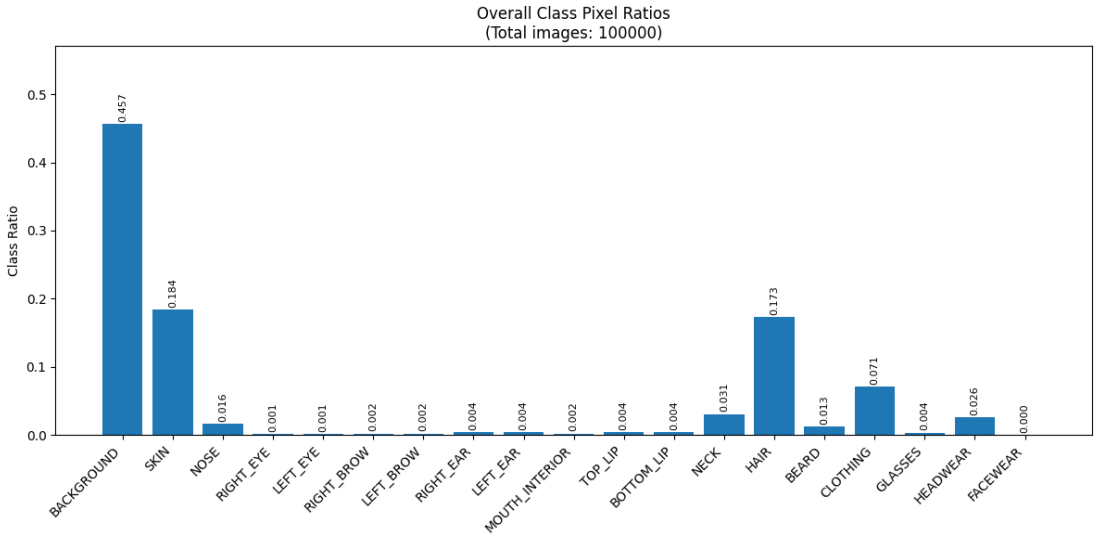


Figure A.1.: Histogram showing percentage of pixels per class label in the dataset.

As illustrated in the Figure A.1, our pixel-wise label distribution analysis of the Face Synthetics dataset reveals the absence of the class (`FACEWEAR` = 18) in the dataset. This

deficiency may adversely affect model performance during inference on real-world data containing such elements.

B. Appendix

This appendix provides a structure comparison (layers, parameters) of models and various encoder variants, offering direction for future in-depth and comprehensive research on different encoder-decoder network architectures.

B.1 Comparison between MaskFormer and Mask2Former

From Table B.1, it can be observed the architectural and strategic differences between Mask2Former and MaskFormer in detail. Mask2Former introduces three major improvements over MaskFormer:

- Transformer Decoder Enhancements: The decoder is modified by reversing the order of self-attention and cross-attention, replacing self-attention with masked attention, introducing learnable query features, and removing dropout for improved efficiency. Masked attention ensures that each pixel attends only to relevant foreground regions, reducing redundant computations.
- Multi-Scale Feature Integration: Features at multiple resolutions from the pixel decoder are fed into corresponding layers of the transformer decoder. Each level is enriched with sine positional encodings and learnable scale embeddings, forming a feature pyramid to improve small object segmentation.
- Efficient Inference via Sampling: Instead of dense computation, mask losses are calculated on a subset of randomly sampled points. Both bipartite matching and final loss use uniform and importance sampling, significantly reducing inference time.

Table B.1.: Comparison between MaskFormer and Mask2Former.

Comparison Item	**MaskFormer**	**Mask2Former**
Encoder in Pixel Decoder	Applies Transformer Encoder (Self-Attn) to the highest downsampled backbone feature	Applies Deformable Transformer Encoder to all three resolution features from the backbone
Memory output from Pixel Decoder	Features output by the above Encoder (derived only from the last layer of backbone features)	Features output by the above Encoder (multi-resolution)
Mask feature output from Pixel Decoder	Fuses the above Memory with backbone features via FPN	Similar to MaskFormer, selects the lowest downsampled feature from Memory and fuses it with backbone features via FPN
Attention in Transformer Decoder	Self-Attn followed by Cross-Attn for Query	Cross-Attn followed by Self-Attn for Query
Mask Attention in Transformer Decoder	None	Core improvement in Mask2Former (see details below)
Target Query	Initialized to zero, non-learnable	Randomly initialized, learnable
Dropout in Transformer Decoder	Present	Removed
Sampling strategy for label assignment and loss calculation	All pixels	Random sampling for label assignment, Uncertainty and Random 3:1 sampling for loss calculation
Mask loss	FocalLoss + DiceLoss	CELoss + DiceLoss

B.2 EfficientNet Variants

Table B.2.: Configuration of EfficientNet Variants (B0–B7).

Model	Input Size (px)	Width Coef.	Depth Coef.	Drop Connect	Dropout Rate
B0	224×224	1.0	1.0	0.2	0.2
B1	240×240	1.0	1.1	0.2	0.2
B2	260×260	1.1	1.2	0.2	0.3
B3	300×300	1.2	1.4	0.2	0.3
B4	380×380	1.4	1.8	0.2	0.4
B5	456×456	1.6	2.2	0.2	0.4
B6	528×528	1.8	2.6	0.2	0.5
B7	600×600	2.0	3.1	0.2	0.5

- drop_connect_rate refers to the dropout ratio applied after the 1×1 projection layer, controlling the probability of randomly dropping connections.
- dropout_rate refers to the dropout ratio applied before the final fully connected (FC) layer in the network.
- width_coefficient represents the scaling factor along the channel dimension. For example, in EfficientNetB0, the 3×3 convolutional layer in Stage 1 uses 32 filters, while in B6 it becomes $32 \times 1.8 = 57.6$, which is then rounded to the nearest multiple of 8, resulting in 56. The same principle applies to other stages.
- depth_coefficient represents the scaling factor along the depth dimension (applicable only from Stage 2 to Stage 8). For instance, in EfficientNetB0, the number of layers in Stage 7 is $L = 4$; in B6, it becomes $4 \times 2.6 = 10.4$, which is rounded up to 11.

B.3 ResNet with Different Depth

Table B.3.: Architectures for Residual Network. Building blocks are shown in brackets, with the number of blocks stacked. Downsampling is performed by conv3_1, conv4_1, and conv5_1 with a stride of 2 [94].

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3,\ 64 \\ 3\times3,\ 64 \end{bmatrix}\times 2$	$\begin{bmatrix} 3\times3,\ 64 \\ 3\times3,\ 64 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times1,\ 64 \\ 3\times3,\ 64 \\ 1\times1,\ 256 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times1,\ 64 \\ 3\times3,\ 64 \\ 1\times1,\ 256 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times1,\ 64 \\ 3\times3,\ 64 \\ 1\times1,\ 256 \end{bmatrix}\times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times3,\ 128 \\ 3\times3,\ 128 \end{bmatrix}\times 2$	$\begin{bmatrix} 3\times3,\ 128 \\ 3\times3,\ 128 \end{bmatrix}\times 4$	$\begin{bmatrix} 1\times1,\ 128 \\ 3\times3,\ 128 \\ 1\times1,\ 512 \end{bmatrix}\times 4$	$\begin{bmatrix} 1\times1,\ 128 \\ 3\times3,\ 128 \\ 1\times1,\ 512 \end{bmatrix}\times 4$	$\begin{bmatrix} 1\times1,\ 128 \\ 3\times3,\ 128 \\ 1\times1,\ 512 \end{bmatrix}\times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times3,\ 256 \\ 3\times3,\ 256 \end{bmatrix}\times 2$	$\begin{bmatrix} 3\times3,\ 256 \\ 3\times3,\ 256 \end{bmatrix}\times 6$	$\begin{bmatrix} 1\times1,\ 256 \\ 3\times3,\ 256 \\ 1\times1,\ 1024 \end{bmatrix}\times 6$	$\begin{bmatrix} 1\times1,\ 256 \\ 3\times3,\ 256 \\ 1\times1,\ 1024 \end{bmatrix}\times 23$	$\begin{bmatrix} 1\times1,\ 256 \\ 3\times3,\ 256 \\ 1\times1,\ 1024 \end{bmatrix}\times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times3,\ 512 \\ 3\times3,\ 512 \end{bmatrix}\times 2$	$\begin{bmatrix} 3\times3,\ 512 \\ 3\times3,\ 512 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times1,\ 512 \\ 3\times3,\ 512 \\ 1\times1,\ 2048 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times1,\ 512 \\ 3\times3,\ 512 \\ 1\times1,\ 2048 \end{bmatrix}\times 3$	$\begin{bmatrix} 1\times1,\ 512 \\ 3\times3,\ 512 \\ 1\times1,\ 2048 \end{bmatrix}\times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

B.4 Swin-Transformer with Different Scales

Table B.4.: Architectures for Swin Transformer[38]. "concat" denotes patch merging operation, "win. sz." indicates window size for self-attention, "dim" is feature dimension, and "head" is the number of attention heads. LN stands for Layer Normalization. The $\times N$ indicates the number of successive transformer blocks with identical configuration.

	downsp. rate (output size)	**Swin-T**	**Swin-S**	**Swin-B**	**Swin-L**
stage 1	$4\times$ 56×56	concat 4×4, 96-d, LN [win. sz. 7×7, dim 96, head 3] $\times 2$	concat 4×4, 96-d, LN [win. sz. 7×7, dim 96, head 3] $\times 2$	concat 4×4, 128-d, LN [win. sz. 7×7, dim 128, head 4] $\times 2$	concat 4×4, 192-d, LN [win. sz. 7×7, dim 192, head 6] $\times 2$
stage 2	$8\times$ 28×28	concat 2×2, 192-d, LN [win. sz. 7×7, dim 192, head 6] $\times 2$	concat 2×2, 192-d, LN [win. sz. 7×7, dim 192, head 6] $\times 2$	concat 2×2, 256-d, LN [win. sz. 7×7, dim 256, head 8] $\times 2$	concat 2×2, 384-d, LN [win. sz. 7×7, dim 384, head 12] $\times 2$
stage 3	$16\times$ 14×14	concat 2×2, 384-d, LN [win. sz. 7×7, dim 384, head 12] $\times 18$	concat 2×2, 384-d, LN [win. sz. 7×7, dim 384, head 12] $\times 18$	concat 2×2, 512-d, LN [win. sz. 7×7, dim 512, head 16] $\times 18$	concat 2×2, 768-d, LN [win. sz. 7×7, dim 768, head 24] $\times 18$
stage 4	$32\times$ 7×7	concat 2×2, 768-d, LN [win. sz. 7×7, dim 768, head 24] $\times 2$	concat 2×2, 768-d, LN [win. sz. 7×7, dim 768, head 24] $\times 2$	concat 2×2, 1024-d, LN [win. sz. 7×7, dim 1024, head 32] $\times 2$	concat 2×2, 1536-d, LN [win. sz. 7×7, dim 1536, head 48] $\times 2$

B.5 SegNeXt-Encoder with Different Scales

Table B.5.: SegNeXt Architecture Specifications [40]. The 'e.r.' stands for the expansion ratio used in the feed-forward network. 'C' refers to the number of channels, while 'L' denotes the number of building blocks. The term 'Decoder dimension' indicates the dimensionality of the MLP used in the decoder.

stage	output size	e.r.	SegNeXt-T	SegNeXt-S	SegNeXt-B	SegNeXt-L
1	$\frac{H}{4} \times \frac{W}{4} \times C$	8	$C = 32, L = 3$	$C = 64, L = 2$	$C = 64, L = 3$	$C = 64, L = 3$
2	$\frac{H}{8} \times \frac{W}{8} \times C$	8	$C = 64, L = 3$	$C = 128, L = 2$	$C = 128, L = 3$	$C = 128, L = 5$
3	$\frac{H}{16} \times \frac{W}{16} \times C$	4	$C = 160, L = 5$	$C = 320, L = 4$	$C = 320, L = 12$	$C = 320, L = 27$
4	$\frac{H}{32} \times \frac{W}{32} \times C$	4	$C = 256, L = 2$	$C = 512, L = 2$	$C = 512, L = 3$	$C = 512, L = 3$
Decoder dimension			256	256	512	1,024
Parameters (M)			4.3	13.9	27.6	48.9