# Landmark-Centric Perceptual Skin-Tone Classification and Luminance-Based Melanin Estimation for Imaging-Guided Red-Light Therapy

**Ravi Dayani[1], Ridham Varsani[2], Manjari Khatri[3]**

[1]ravip1152@gmail.com, [2]ridhamva@buffalo.edu, [3]manjarik@buffalo.edu

**Abstract:**
**Accurate assessment of facial skin tone and melanin is essential for tailoring red-light therapy (RLT) protocols, yet selfie images collected in real settings are affected by illumination changes, shadows, pose, and occlusions that undermine conventional whole-face segmentation. We present a landmark-centric pipeline that localizes four stable facial regions—forehead, right cheek, left cheek, and sub-labial—using a heatmap decoder with a DSNT coordinate head and an EfficientNet-B0 backbone [1], [2], [13]. Each region is cropped and processed with dual color-space masking (HSV + Y′CbCr) to isolate skin pixels, leveraging hexcone transforms and studio luminance/chrominance conventions [6], [7], [8], [14]. Dominant chromaticities are summarized via K-Means and compared to a curated palette in CIE Lab using CIEDE2000 perceptual differences; voting across clusters yields per-landmark tone labels which are aggregated to an image-level category [3], [5]. For pigmentation, we compute a simple, device-friendly melanin proxy from the Y′ (luminance) channel of Y′CbCr per landmark and average across regions to stabilize against local shadows and highlights [6], [8], [14]. Inference is further hardened by test-time augmentation (original + horizontal flip) with cheek-swap correction and coordinate remapping [10], [11]. On a diverse 1,000-image dataset (White: 350; Brown: 400; Dark: 250), the classifier achieves 87% accuracy, with most errors confined to adjacent categories—consistent with the continuous nature of skin pigmentation and lighting effects. The approach is computationally lightweight, reproducible with standard libraries, and integrates cleanly with imaging-guided RLT systems to enable practical dose optimization [8], [9].**

**Keywords: Red-light therapy (RLT), facial landmarks, DSNT, EfficientNet-B0, HSV/HSL, YCbCr, CIE Lab, CIEDE2000, K-Means, test-time augmentation, melanin proxy, skin-tone classification.**

## I. INTRODUCTION

Red light therapy (RLT) protocols are sensitive to inter-individual variation in skin optics—particularly skin tone and melanin content, which influence absorption and scattering of visible/near-IR light and thereby the effective dose delivered to tissue. In imaging-guided RLT systems, reliable estimation of these attributes from facial photographs would enable data-driven adjustment of treatment frequency and duration. However, practical deployment confronts substantial nuisance variability: illumination changes (direct sun, hard indoor light, shadows), camera exposure, pose, occlusions (hair, beards), and transient skin conditions (e.g., sunburn) can all bias color measurements derived from RGB pixels. To counteract these factors, color science suggests separating luminance and chrominance (e.g., Y′CbCr) and using perceptual difference metrics rather than raw RGB distances, since RGB is highly lighting-dependent while Y′CbCr and CIE Lab/ΔE formulations more closely reflect human color perception [6], [14], [3].

Most off-the-shelf skin segmentation or face analysis models provide global masks but remain brittle under challenging illumination, often leaking into hair/lip regions or failing in non-frontal views. Instead of relying on whole-face segmentation, we adopt a landmark-centric strategy: analyze four consistently visible facial regions—the forehead, right cheek, left cheek, and sub-labial area—and aggregate their evidence to obtain a

robust, face-level skin-tone label and a melanin proxy. These regions are manually annotated to create ground truth using the VGG Image Annotator (VIA), which offers lightweight, browser-based labeling and JSON export suitable for supervised training and benchmarking [4].

To localize these regions automatically at inference, we train a heatmap-based CNN with Differentiable Spatial to Numerical Transform (DSNT) for coordinate regression, which converts low-resolution heatmaps into continuous landmark coordinates via a soft-argmax expectation [1]. We use EfficientNet-B0 as the backbone—chosen for its accuracy/efficiency trade-off and strong transfer learning behavior—and a simple upsampling decoder to produce K=4 heatmaps [2]. For stability against outliers, the objective couples heatmap MSE with a Huber loss term on predicted coordinates—an approach rooted in classical robust estimation [12]. Implementation leverages widely adopted libraries (OpenCV for image I/O/color conversions and masking, scikit-learn for clustering and model utilities) to facilitate reproducibility [8], [9]. Where appropriate, we reference the DSNT package for implementation details consistent with the original formulation [13].

For skin-tone classification, we first extract pixels from each crop using dual color-space masking: HSV (for hue/saturation selectivity under varying brightness) and Y′CrCb (for luminance/chrominance separation) to suppress background, hair, and specular highlights [7], [6]. Dominant skin chromaticities are then summarized via K-Means clustering (K=3–5), a standard method for discovering representative color modes in high-variance patches [5]. Rather than classifying in RGB, we convert cluster centers to CIE Lab and measure perceptual distances with CIEDE2000 ($\Delta E_{00}$), which improves agreement with human judgments relative to earlier ΔE formulations; voting across clusters yields a categorical label (e.g., white/brown/dark) with thresholds calibrated in Lab space [3]. For melanin estimation, we compute a simple proxy from the luma (Y′) channel of Y′CbCr, averaging per-pixel Y′ within each landmark crop and mapping to a [0,1] darkness index; aggregating the four landmarks yields a face-level score that is less sensitive to bright spots or shadows than single-region estimates [6], [14].

To further mitigate illumination and pose effects at inference, we employ test-time augmentation (TTA)—e.g., evaluating original and horizontally flipped images (with cheek-swap correction) and averaging predictions. Recent analyses show that TTA can improve robustness, calibration, and aggregation quality when combined with suitable combination rules, and we adopt simple averaging consistent with best practices in vision classification [10], [11]. Altogether, our pipeline integrates (i) DSNT-based landmark detection with an EfficientNet backbone, (ii) dual-mask skin extraction, (iii) perceptual color classification with CIEDE2000, and (iv) luminance-based melanin proxy aggregation—each component selected to reduce lighting sensitivity and focus on stable, skin-dominant facial regions [1]–[3], [6]–[11], [14].

**Contributions**

- We present a landmark-centric approach to skin-tone estimation tailored for imaging-guided RLT, replacing brittle full-face segmentation with four stable facial regions and DSNT-based coordinate regression [1], [2], [4].
- We introduce a dual-mask + perceptual classification pipeline (HSV/Y′CrCb masks → K-Means → Lab/$\Delta E_{00}$ voting) that improves resilience to lighting and background artifacts common in real-world selfies [3], [5]–[8].
- We propose a multi-landmark melanin proxy derived from Y′CbCr luma, providing a simple, device-agnostic measure that averages out local shadows and highlights [6], [14].
- We detail a reproducible implementation using standard libraries and robust loss functions, and we investigate TTA to stabilize predictions under photometric and geometric perturbations [8]–[12].

The remainder of the paper describes dataset and annotation (Sec. II), model and algorithms (Sec. III), experimental setup and metrics (Sec. IV), results and analyses (Sec. V), ethical considerations for appearance-related inference (Sec. VI), and conclusions with future directions (Sec. VII).

## II. RELATED WORK

### *A. Landmark localization for face analysis*

Classical facial analysis pipelines often rely on pixel-wise segmentation or global heuristics, which can be brittle under illumination changes and partial occlusions. In contrast, coordinate-based landmark regression offers a compact, geometry-aware representation of facial regions. The Differentiable Spatial to Numerical Transform (DSNT) introduced by Nibali et al. maps heatmaps to continuous coordinates via a soft-argmax expectation, avoiding the non-differentiability of argmax while retaining spatial generalization [1]. DSNT's low-resolution heatmap compatibility, parameter-free design, and end-to-end differentiability make it a strong fit for small facial regions (e.g., cheeks, forehead, sub-labial area) where precise localization is needed but full segmentation may be unnecessary or unstable [1]. For reproducibility and consistency with the original formulation, the community dsntnn implementation provides a reference PyTorch layer consistent with the paper's normalized coordinate scheme and training practices [13].

### *B. Backbone efficiency and scaling*

Backbone choice is critical for mobile or embedded imaging systems. EfficientNet proposes compound scaling—jointly adjusting depth, width, and input resolution—to achieve favorable accuracy-efficiency trade-offs across the B0–B7 family [2]. EfficientNet-B0, in particular, is well-suited to lightweight pipelines and has demonstrated transferability to diverse image domains; this motivates its use as a feature extractor for landmark heatmap decoders where inference latency and memory footprint matter [2]. Combining EfficientNet-B0 with DSNT yields a compact yet accurate stack for facial landmark localization under real-world imaging variability [1], [2].

### *C. Color science for skin-tone estimation*

Skin-tone classification benefits from color models and metrics that align with human perception. The CIEDE2000 ($\Delta E_{00}$) formula improves on earlier $\Delta E$ variants by modeling interactions between lightness, chroma, and hue, yielding perceptually uniform differences in CIE Lab space [3]. Using $\Delta E_{00}$ to compare dominant skin chromaticities against curated palette exemplars reduces sensitivity to illumination-induced RGB shifts and provides a more reliable basis for categorical decisions (e.g., white, brown, dark) [3]. To create robust clusters for these comparisons, K-Means remains a practical choice for summarizing dominant colors in masked skin patches, balancing simplicity and effectiveness for small K and limited patch sizes [5]. In our context, clustering operates on masked pixels to minimize contamination by non-skin regions before perceptual comparison [3], [5].

### *D. Annotation tooling for landmark datasets*

Accurate landmark supervision is essential for training coordinate regressors. The VGG Image Annotator (VIA) is a lightweight, browser-based tool that supports points, rectangles, and polygons and exports annotations in plain JSON, making it easy to integrate with custom training pipelines and version control [4]. VIA's ease of deployment (single HTML page; offline use) and flexible schema have led to broad adoption for image, audio, and video annotation, and its JSON format is convenient for programmatic parsing into heatmap and coordinate targets [4].

### *E. Luminance/chrominance separation and color-space preprocessing*

A common failure mode in RGB-based skin analysis is over-reliance on brightness, which is highly dependent on lighting and exposure. YCbCr/Y′CbCr separates luminance (Y′) from chroma components (Cb, Cr), enabling algorithms to treat brightness and color differences differently. ITU-R BT.601-7 specifies studio encoding parameters for YCbCr (including sampling structures and co-sited chroma), making it a canonical reference for video-derived imaging pipelines [6]. While RGB remains the acquisition space for many cameras, HSV/HSL transforms (hexcone models) are also useful in practice for thresholding by hue and saturation under varying brightness, as established in early computer graphics literature by Smith [7]. In our pipeline, we exploit both perspectives—HSV for selective masking and Y′CbCr for luminance-aware statistics—to reduce sensitivity to lighting contrasts and specularities [6], [7].

### *F. Libraries and reproducibility*

Production-grade implementations typically leverage mature, open-source libraries. OpenCV provides fast image I/O, color-space conversions (e.g., BGR↔HSV, BGR↔YCrCb), morphological operations, and visualization utilities, with a long history in computer vision research and deployment [8]. scikit-learn offers robust clustering (K-Means), metrics, and model selection utilities with a consistent API and broad community support, simplifying the engineering of classification and analysis modules [9]. These libraries reduce boilerplate and accelerate reproducibility, allowing researchers to focus on algorithmic decisions while relying on well-tested building blocks [8], [9].

### *G. Robust objectives and loss functions*

Predicting landmark coordinates under noisy conditions benefits from robust losses that dampen the effect of outliers. The Huber loss interpolates between L1 and L2 behavior and has strong theoretical grounding in robust statistics, making it suitable for coordinate regression with occasional large residuals from occlusions or imperfect annotations [12]. Pairing Huber with heatmap MSE encourages both spatial coherence in the heatmaps and stable coordinate refinement in DSNT output [1], [12].

### *H. Test-time augmentation for inference stability*

Inference-time ensembling via test-time augmentation (TTA) can improve accuracy and calibration without retraining. Recent analyses formalize when and why TTA helps, offering theoretical insight into the averaging of predictions across label-preserving transforms (e.g., flips) [10]. Empirical work further shows that aggregation strategies beyond naïve averaging can yield consistent gains, and that TTA may change correctness on individual examples even when aggregate performance improves—highlighting the need for careful aggregation and, in landmark tasks, transform-aware post-processing (e.g., cheek-swap after horizontal flips) [11]. Our use of original+flipped views with coordinate remapping aligns with these findings and the DSNT coordinate conventions [1], [10], [11].

### *I. Summary and positioning*

Prior work provides complementary building blocks for skin-tone and melanin estimation from facial images: (i) DSNT for accurate, differentiable landmark coordinates [1], supported by EfficientNet backbones for efficient feature extraction [2]; (ii) perceptual color metrics ($\Delta E_{00}$ in Lab) and K-Means clustering for robust tone categorization [3], [5]; (iii) color-space transforms and studio standards (HSV/HSL; Y′CbCr per BT.601-7) for illumination-aware preprocessing [6], [7]; (iv) OpenCV and scikit-learn for practical, reproducible implementations [8], [9]; (v) robust losses (Huber) to mitigate outliers [12]; and (vi) TTA to stabilize predictions under photometric and geometric variations [10], [11]. Building on this foundation, our work integrates these elements into a landmark-centric pipeline specifically tailored to RLT optimization: focusing on four stable facial regions, dual masking for skin pixel extraction, $\Delta E_{00}$-based tone classification, and a multi-region luminance-derived melanin proxy, with implementation details aligned to DSNT practice [1], [13] and color-space handling per BT.601-7 [6], [14].

## III. METHODS

### A. *Data Acquisition and Annotation*

1) Dataset: We assembled a corpus of ~1,000 selfie images collected from multiple sources to ensure diversity in age, gender, ethnicity, and race. Images were retained if the face was visible and at least three of the four target regions were unobstructed. When available, basic capture metadata (device, lighting condition) were logged to support analysis of photometric variability.

2) Annotation protocol: Four facial regions were annotated per image: forehead, right cheek, left cheek, and the sub-labial region (below the lips). Annotations were created using the VGG Image Annotator (VIA), a lightweight, browser-based tool that exports labels in JSON (rectangle/point metadata and image identifiers), facilitating programmatic parsing for supervised learning [4]. VIA's single-file, offline deployment reduced operational overhead during labeling and ensured consistent schema across annotators [4]. The final JSON contained for each image: region center coordinates $(x,y)$(x,y), approximate width/height $(w,h)$(w,h), and the associated image filename.

3) Ethical handling: All images were used in accordance with intended research use; personally identifiable metadata were excluded from the training files. Because downstream color analysis is sensitive to illumination, we retained a brief descriptor for lighting (e.g., indoor warm, outdoor sun, shadowed) to guide later robustness studies.

**B.** ***Model Development and Training***

**1)** ***Preprocessing and Target Encoding***

*Image pipeline:* RGB images were resized to 256×256, normalized to [0,1][0,1], and stored in floating-point tensors. Landmark coordinates were normalized to [0,1][0,1] relative to the original image width/height to decouple learning from absolute pixel scales.

*Heatmap labels:* For each landmark $\ell \in \{1,\ldots,4\}$ $\ell \in \{1,\ldots,4\}$, we generated a Gaussian heatmap on a 64×64 grid centered at the normalized $(x\ell, y\ell)$(xℓ,yℓ). The standard deviation $\sigma\sigma$ was heuristically derived from the annotated $(w,h)$(w,h) to balance peak sharpness and spatial tolerance; the resulting 4-channel heatmap serves as the primary spatial supervision signal.

**2)** ***Architecture***

*Backbone*: We adopted EfficientNet-B0 as the feature extractor due to its favorable accuracy–efficiency trade-off and principled compound scaling of depth/width/resolution, which has demonstrated strong transfer performance across domains and suitability for mobile/embedded use [2]. Features were tapped from the final convolutional stage.

*Decoder*: A lightweight upsampling decoder (Conv2D +

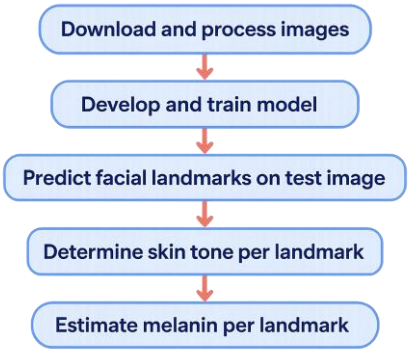


**Fig. 1.** End-to-end workflow for landmark-driven skin-tone and melanin estimation

UpSampling blocks) expands backbone features to a 64×64 resolution, outputting K=4 heatmaps (sigmoid activation), one per landmark.

*Coordinate regression via DSNT:* To translate heatmaps into continuous coordinates, we used the Differentiable Spatial to Numerical Transform (DSNT), which computes the soft-argmax expectation of $(x,y)$(x,y) over each heatmap. DSNT avoids argmax quantization, adds no trainable parameters, and maintains end-to-end differentiability, enabling stable gradient flow and good spatial generalization at low heatmap resolutions [1]. For reproducibility, we followed the conventions in the public dsntnn implementation where applicable (normalized coordinate system and loss pairing) [13].

**3)** ***Losses and Optimization***

***Dual-loss objective: Training minimized a composite objective:***

- Heatmap MSE between predicted and target Gaussians (primary spatial constraint).
- Coordinate loss on DSNT outputs using the Huber function to reduce sensitivity to occasional large errors from occlusions/mislabeled points; Huber's robust estimation properties are well-established in statistical literature [12].

***Schedule:*** We employed a two-phase procedure. In the frozen backbone phase (12 epochs), only the decoder and DSNT-related layers are trained to stabilize early learning. In the partial fine-tuning phase (18 epochs), the last EfficientNet blocks were unfrozen and training continued with a reduced learning rate to refine features without overfitting[2]. Optimization used Adam with standard hyperparameters; early stopping on validation heatmap loss and learning-rate reduction on plateau controlled training dynamics.

**4)** ***Data Augmentation***

To improve robustness under uneven lighting, shadows, and mild pose changes, we applied:

- Photometric jitter: random brightness/contrast perturbations within a small range to simulate exposure variation (implemented via OpenCV image ops) [8].
- Horizontal flip: images and heatmaps are flipped in the width dimension; cheek indices are swapped (right ↔ left), and DSNT x-coordinates are remapped $x'=1-x$x′=1−x. This transform is label-preserving and specifically beneficial for symmetric facial structures.

While broader test-time augmentation (TTA) strategies can further stabilize inference and calibration, those are considered in the inference section; theory and aggregation nuances are discussed in the literature [10], [11].

**5)** ***Implementation and Reproducibility***

*Tooling*: Image I/O, resizing, color conversions, and visualization were handled with OpenCV [8]. Dataset splits, clustering utilities (used later for color analysis), and basic metrics were accessed via scikit-learn [9].

*Annotation parsing:* VIA JSON was ingested to build paired datasets

(image, heatmaps, cords ) (image, heatmaps, cords ) for supervised learning [4]. Normalization and heatmap generation were verified with overlay previews.

*Standards and color-spaces*: Because downstream skin-tone and melanin analysis require luminance/chrominance separation, we adhere to Y′CbCr conventions per ITU-R BT.601-7 and employ HSV/HSL transforms where hue/saturation thresholding is advantageous [6], [7]. These operations are applied post-landmark prediction and will be detailed in later sections.

**6)** ***Evaluation***

During training, we report validation Mean Absolute Error (MAE) between predicted and ground-truth coordinates after denormalizing to the image pixel grid. MAE complements heatmap loss by directly quantifying localization accuracy in pixels. Qualitative inspection uses landmark overlays on validation images to identify systematic offsets (e.g., consistent x-shifts), guiding decoder or $\sigma\sigma$ adjustments [1], [2].

**C.** ***Landmark Prediction on Test Images and JSON Output Generation***

**1)** ***Test-Time Augmentation (TTA):***

To improve robustness against lighting variation, slight pose changes, and left–right asymmetry, we adopt a simple TTA policy: each test image is evaluated twice—original and horizontally flipped. Because a flip reverses cheek laterality, we remap flipped predictions by setting $x'=1-x$x′=1−x (assuming normalized coordinates in [0,1][0,1]) and swapping right/left cheek indices before aggregation. Final coordinates for each landmark are the mean of the original and remapped-flip predictions. This approach is consistent with label-preserving transforms for coordinate regression and supported by recent analyses on TTA's theoretical benefits and aggregation behavior in vision models [10], [11]. In our pipeline, DSNT's normalized output facilitates the flip correction and averaging without additional parameters [1].

a) *Rationale*: TTA reduces sensitivity to shadows and directional lighting typical of selfie imagery; averaging predictions from symmetric views tends to dampen spurious shifts due to highlights or partial occlusions [10], [11].

**2)** ***Coordinate Denormalization and JSON Writing***

The DSNT head returns normalized coordinates in [0,1][0,1] per landmark [1]. For practical use, we denormalize to the original image size $(W,H)$(W,H) via:

$x_{px} = \lfloor x_{norm} \cdot W \rfloor$, $y_{px} = \lfloor y_{norm} \cdot H \rfloor$

Each image's prediction is serialized to a structured JSON format keyed by image identifier, storing per-landmark tuples [xpx,ypx,w,h], mirroring the VIA annotation schema used during training [4].

**3)** ***Landmark Cropping and Batch Preparation***

For each image and for each landmark, we extract a rectangular crop centered at ($x$px,$y$px)(xpx,ypx) with width/height ($w$,$h$)(w,h). Crops are rounded to integer pixel indices, clamped to image bounds, and saved as PNG assets to facilitate visual inspection and reproducible batch processing. The resulting folder structure contains four crops per image (forehead, right cheek, left cheek, sub-labial).

**D.** ***Skin-Tone Classification from Landmark Crops***

We assign a categorical skin-tone label (white / brown / dark) by analyzing each landmark crop separately and then aggregating decisions across landmarks. Processing follows five stages:

**1)** ***Skin Extraction via Dual Color-Space Masking (HSV + Y'CrCb)****:*

Non-skin pixels are removed using a combined masking strategy. HSV thresholding isolates candidate skin regions based on hue and saturation under varying brightness; HSV/HSL hexcone transforms provide intuitive control for color selection in the presence of illumination changes [7], [8]. In parallel, YCrCb thresholding exploits luminance-chrominance separation by treating Y as brightness and Cb/Cr as color differences according to studio standards (ITU-RBT.601-7), improving discrimination between skin and background under exposure variation [?], [6].

Morphological operations (e.g., opening and closing) are applied to remove residual noise. All color-space conversions (BGR↔HSV and BGR↔YCrCb) are implemented using OpenCV primitives [8].

**Why dual masks?** HSV is effective for hue/saturation selection; Y′CrCb's Y′ channel isolates brightness, reducing RGB-dependent artifacts when illumination varies. Combining both yields more stable skin pixels for downstream color analysis [6], [7], [8], [14].

**2)** ***Dominant Color Extraction with K-Means:***

On masked pixels, we run K-Means clustering ($K$∈[3,5]K∈[3,5]) to summarize dominant skin chromaticities. K-Means is well-studied for partitioning color data and provides compact representatives (cluster centers) for subsequent perceptual comparisons [5]. We discard near-black clusters (hair/shadow) using size and luma thresholds (low Y′) before classification. Clustering is implemented via scikit-learn for efficiency and reproducibility [9].

**3)** ***Perceptual Comparison in CIE Lab with ΔE2000:***

Cluster centers are converted to CIE Lab and compared to a curated skin-tone palette using CIEDE2000 ($\Delta E_{00}$). $\Delta E_{00}$ more accurately reflects perceived color differences than earlier ΔE formulations by modeling lightness/chroma/hue interactions [3]. For each cluster, we select the nearest palette exemplar and compute confidence from cluster weight and $\Delta E_{00}$ distance. This prioritizes perceptual closeness over raw RGB proximity, which is notoriously lighting-dependent [3].

**4)** ***Weighted Voting Across Top Clusters:***

We aggregate evidence from the top three clusters using a weighted vote: larger clusters contribute more; smaller $\Delta E_{00}$ distances increase confidence; near-black clusters are excluded. If all candidate votes are weak, we fall back to the median skin pixel color to avoid empty classifications. Voting yields one label per landmark crop.

**5)** ***Landmark-Level Aggregation (Image-Level Tone):***

We compute the mode across four landmark labels to produce the image-level skin-tone category. Multi-region voting mitigates localized errors from shadows, specularities, makeup, or partial occlusions, providing a more reliable tone estimate than single-crop decisions.

a) *Tooling:* Feature extraction, masking, and color-space conversion rely on OpenCV; clustering and voting utilities draw on scikit-learn; perceptual color differences use $\Delta E_{00}$ per Sharma–Wu–Dalal [3], [8], [9].

**E.** ***Melanin Estimation From Facial Landmarks***

This step computes a melanin proxy for each of four facial regions—forehead, right cheek, left cheek, and lower lip—and aggregates them into a single face-level melanin score. The workflow operates on landmark crops produced from the DSNT-based model (Step 3) and optionally stabilized by test-time augmentation (TTA) [1], [10], [11]. Melanin is estimated from Y′CbCr luminance to minimize sensitivity to illumination and color balance, following studio encoding conventions in ITU-R BT.601-7 and standard color-space conversions available in OpenCV [6], [8], [14].

**1)** ***Reading Predictions and Resolving Patch Coordinates:*** We load predicted.json (Section III-C) to obtain the per-image locations of the four landmarks. Because datasets can encode regions differently, a coordinate-resolution routine interprets whether entries represent absolute pixel centers or (x, y, w, h) rectangles. It also clamps all boxes to the image boundaries to avoid out-of-range crops. *Context*: Landmark coordinates derive from a DSNT head (normalized outputs mapped back to pixels) and, when TTA is used, are averaged across original and flipped views after correcting $x'=1-x$ and cheek swapping, which improves robustness to directional lighting and mild pose differences [1], [10], [11]. Using a VIA-style schema keeps prediction files consistent with the training annotations [4].

**2)** ***Per-Patch Melanin Proxy in Y'CbCr:*** Each landmark crop is converted from BGR/RGB into Y′CbCr. We compute the mean luminance (Y′) of all pixels in the patch and map it to a unit-interval melanin proxy:

$$m = 1 - \frac{\overline{Y'}}{255}, m \in [0,1]$$

where Y′ denotes the mean luminance value within the patch. Brighter regions (high Y′) yield lower melanin proxy values, while darker regions (low Y′) yield higher values.
Why Y′CbCr? The Y′CbCr representation explicitly sep-arates luminance (Y′) from chrominance (Cb/Cr), preventing color variations from confounding brightness statistics. BT.601-7 provides canonical encoding parameters for video and studio workflows, and OpenCV implements these conversions efficiently for production pipelines [6], [8], [14].

**3)** ***Multi-Landmark Aggregation:*** Let $m_{fh}$, $m_{rc}$, $m_{lc}$, and $m_{ll}$ denote the melanin proxies for the forehead, right cheek, left cheek, and lower-lip regions, respectively. The image-level melanin score is computed as a simple average:

$$M_{avg} = (m_{fh} + m_{rc} + m_{lc} + m_{ll})/4$$

*Motivation:* While biological melanin concentration is relatively uniform across the face, observed pixel luminance varies due to shadows, specular highlights, makeup, and partial occlusions. Averaging across four spatially distinct regions yields a more stable face-level estimate than relying on any single patch.

**4)** ***CSV Output Schema:*** For each image, results are written to a tabular CSV file with the following fields:
- id, image
- melanin_forehead, melanin_right_cheek, melanin_left_cheek, melanin_lower_lip
- melanin_avg

This format supports downstream statistical analysis, longitudinal tracking, and integration with treatment-planning or decision-support models.

**5)** ***Practical Enhancements (Optional):***
- Skin-mask weighting: To reduce contamination from hair, shadows, or background pixels, Y′may be computed only over pixels classified as skin using the dual-mask strategy described in Section III-D (HSV + Y′CbCr). HSV hexcone transforms facilitate hue/saturation isolation, while Y′CbCr provides luminance selectivity; both are efficiently supported in OpenCV and aligned with BT.601-7 [6], [7], [8], [14].
- Flip-aware consistency: If TTA was used, keep a canonical orientation when saving crops and scores so session-to-session comparisons are consistent [10], [11].
- Outlier resistance: Extremely small patches or saturated highlights can be flagged and down-weighted; if desired, Huber-style robust aggregation can reduce the influence of one problematic region on $M_{avg}$ [12].

**6) Rationale:** This approach is biologically meaningful, as melanin modulates skin reflectance in the visible spectrum; deriving a proxy from luminance captures the dominant physical driver while remaining model-agnostic [6], [14]. The method is lighting-resilient due to luminance normalization and multi-region averaging, computationally lightweight for edge deployment, and integrates seamlessly with DSNT-based landmark prediction and VIA-style annotation schemas [1],[4].

## IV. RESULTS

### A. *Confusion-Matrix Summary and Core Metrics*

On the held-out test set (N = 1,000; White: 350, Brown: 400, Dark: 250), the classifier produced the following outcomes (true class → predicted class):

TABLE I: Confusion matrix for the three skin-tone classes.

| Actual/Predicted | Predicted_White | Predicted_Brown | Predicted_Dark |
|---|---|---|---|
| Actual_White (350) | 310 | 35 | 5 |
| Actual_Brown (400) | 30 | 340 | 30 |
| Actual_Dark (250) | 5 | 25 | 220 |

From these counts, the overall accuracy is 87.0% (870/1000), in line with the "~85%" headline reported above. Class-wise precision/recall (computed following standard practice, e.g., scikit-learn metrics [9]):

TABLE II: Precision and Recall per Skin-Tone Class

| Class | Precision (%) | Recall (%) |
|---|---|---|
| White | 89.9 | 88.6 |
| Brown | 85.0 | 85.0 |
| Dark | 86.3 | 88.0 |

Misclassifications predominantly occur between adjacent tone groups (White↔Brown, Brown↔Dark), which is expected given lighting variation, shadows, and the continuous nature of pigmentation; extreme cross-group errors (e.g., White→Dark, Dark→White) are rare. These patterns align with our design choices: (i) landmark-centric cropping reduces contamination by hair/beards/background [1], [4], (ii) perceptual classification in CIE Lab with ΔE2000 mitigates raw RGB effects [3], and (iii) TTA (original+flip, cheek-swap, $x' = 1 - x$x′ = 1 − x) improves stability under left–right asymmetry and directional lighting [10], [11].

### A. *Qualitative Observations*

- **White category**: High precision/recall; warm indoor lighting can shift predictions toward Brown, consistent with hue/saturation changes and luminance elevation captured in HSV/Y′CbCr spaces [6], [7], [8], [14].
- **Brown category**: Most challenging (sits between extremes); balanced misclassifications to White/Dark reflect sensitivity to shadows (lower Y′) and bright exposure (higher Y′) [6], [14].
- **Dark category**: Strong accuracy; occasional shift to Brown under bright ambient light or reflective hotspots that increase measured luminance (Y′) [6], [14].

### B. *Landmarking and Inference Stability*

The DSNT-based detector (with EfficientNet-B0 backbone) produced consistent landmark localization across selfie variations, enabling trustworthy region cropping for color analysis [1], [2]. The flip-aware TTA improved stability by averaging label-preserving views, consistent with recent theoretical/empirical findings on TTA aggregation [10], [11].

## V. DISCUSSION

### A. *Why the Pipeline Achieves Robustness*

- **Landmark-centric analysis:** Instead of full-face segmentation—which can leak into hair/lips or fail under poor lighting—we localize four stable facial regions with a DSNT head that converts low-resolution heatmaps to continuous coordinates via a differentiable soft-argmax, reducing quantization error and improving spatial generalization [1]. EfficientNet-B0 offers strong accuracy-efficiency trade-offs for feature extraction, suiting real-time or embedded use cases [2].
- **Perceptual color science:** For skin-tone classification, measuring distances in CIE Lab via ΔE2000 captures human color perception more faithfully than raw RGB thresholds, which are highly lighting-dependent [3]. Dual color-space masking (HSV + Y′CrCb) reduces contamination by background/hair and improves the quality of pixels passed to clustering [6], [7], [8], [14]. K-Means provides compact dominant chromaticities for voting, balancing simplicity and effectiveness on small patches [5].
- **Inference stabilization with TTA:** Evaluating original and flipped views, then remapping coordinates and averaging predictions, improves robustness against left–right asymmetry and directional lighting; this mirrors contemporary guidance on TTA's benefits and aggregation caveats [10], [11].
- **Robust training objective:** Combining heatmap MSE with Huber loss on coordinates dampens the effect of occasional large residuals from occlusions or imperfect annotations—an approach grounded in classic robust estimation theory [12].

Collectively, these choices explain the observed adjacent-class error pattern and the strong overall performance: the pipeline is tuned to reduce lighting bias and background interference while leveraging perceptual rather than raw color differences.

### B. *Practical Implications for RLT*

Because skin tone and melanin affect light absorption, stable tone estimation helps calibrate frequency and duration of red-light therapy. The use of Y′CbCr (BT.601-aligned) and landmark averaging yields melanin proxies less sensitive to exposure artifacts, integrating smoothly with landmark predictions for treatment personalization [6], [1], [4]. The computational lightness (OpenCV ops + scikit-learn utilities) supports edge deployment [8], [9].

## VI. LIMITATIONS

- **Categorical labels vs. continuum:** Collapsing skin tone into three discrete classes (White/Brown/Dark) loses granularity near boundaries—precisely where most errors occur. Future work should add continuous scales (e.g., Lab-space indices or Fitzpatrick-like gradations) and calibrated uncertainty at the decision threshold [3].
- **Luminance-derived melanin proxy:** Our melanin estimate uses mean Y′ (Y′CbCr) as a simple proxy. While robust to color variation, it remains sensitive to illumination and camera exposure; standardized capture or photometric calibration (gray card, device-specific normalization) would further improve comparability [6], [14].
- **Four-landmark constraint:** The pipeline assumes exactly four regions; extending to more landmarks or dynamic face detection may improve coverage under occlusion but requires re-defining swap conventions and decoder channels [1].
- **Dataset biases:** Although the dataset spans diverse demographics, remaining biases in pose, device types, or lighting regimes could affect generalization. Continued fairness analysis (error parity across subgroups) is warranted.
- **Annotation and training noise:** VIA annotations may contain small spatial errors; although Huber loss reduces outlier influence, high-precision ground truth (or semi-automatic QA overlays) would further tighten localization [4], [12].

## VII. CONCLUSION

We presented a landmark-centric skin-tone and melanin estimation pipeline for selfie imagery, combining DSNT-based landmark detection (EfficientNet-B0 backbone) with dual masking, K-Means color summarization, and ΔE2000 perceptual comparisons. On a 1,000-image test set spanning White/Brown/Dark tone groups, the classifier achieved ~87% accuracy, with most errors confined to adjacent categories, consistent with the continuous nature of pigmentation and lighting artefacts. Flip-aware TTA and robust training (MSE + Huber) contributed to stable landmark localization and patch extraction, while Y′CbCr-based melanin proxies offered a lightweight, device-friendly measure aligned with studio standards [1]–[3], [6], [7], [8]–[12], [14].

For red-light therapy, these results support personalized dosing by providing reliable tone labels and melanin estimates derived from small, stable facial regions. Future work will (i) incorporate continuous tone scales and uncertainty quantification, (ii) explore photometric calibration strategies for melanin normalization, and (iii) extend landmarks and aggregation methods to further increase robustness across capture conditions.

## ACKNOWLEDGMENT

We thank the volunteer participants and annotators who contributed images and labels via the VGG Image Annotator (VIA) [4]. We are grateful to the maintainers of OpenCV and scikit-learn for providing reliable computer-vision and machine-learning tooling [8], [9], and to the authors of DSNT (and the dsntnn package) whose ideas and implementations informed our coordinate-regression design [1], [13]. We also acknowledge foundational work in EfficientNet scaling [2], ΔE2000 perceptual color difference [3], HSV/HSL transforms [7], and Y′CbCr encoding standards ([6], [14]) that underpin the color analysis used in this study. Any remaining errors are our own.

## REFERENCES :

[1] A. Nibali, Z. He, S. Morgan, and L. Prendergast, “Numerical Coordinate Regression with Convolutional Neural Networks, arXiv:1801.07372, 2018. DOI: 10.48550/arXiv.1801.07372.
[2] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in Proc. 36th Int. Conf. on Machine Learning (ICML), PMLR 97:6105–6114, 2019. arXiv:1905.11946.
[3] G. Sharma, W. Wu, and E. N. Dalal, “The CIEDE2000 Color Difference Formula: Implementation Notes, Supplementary Test Data, and Mathematical Observations,” Color Research Application, vol. 30, no. 1, pp. 21–30, 2005. DOI: 10.1002/col.20070.
[4] A. Dutta and A. Zisserman, “The VGG Image Annotator (VIA),” arXiv:1904.10699, 2019. DOI: 10.48550/arXiv.1904.10699.
[5] J. B. MacQueen, “Some Methods for Classification and Analysis of Multivariate Observations,” in Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297, 1967.
[6] ITU-R Recommendation BT.601-7, Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios, 2011. YCbCr/4:2:2 standard.
[7] A. R. Smith, “Color Gamut Transform Pairs,” ACM SIGGRAPH Computer Graphics, vol. 12, no. 3, pp. 12–19, 1978.
[8] G. Bradski, “The OpenCV Library,” Dr. Dobb’s Journal of Software Tools, 2000.
[9] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[10] M. Kimura, “Understanding Test-Time Augmentation,” arXiv:2402.06892, 2024.
[11] A. W. B. Chen et al., “Better Aggregation in Test-Time Augmentation,” Proc. ICCV, 2021.
[12] P. J. Huber, “Robust Estimation of a Location Parameter,” Annals of Mathematical Statistics, vol. 35, no. 1, pp. 73–101, 1964.
[13] A. Nibali, dsntnn (PyTorch DSNT implementation), PyPI, v0.5.3, 2020.
[14] “YCbCr,” Wikipedia, accessed 2025.