Enhanced K-means Color Clustering Based on SLIC Superpixels Merging incorporated within the Entomology Software: AInsectID

Enhanced K-means Color Clustering Based on SLIC Superpixels Merging incorporated within the Entomology Software: AInsectID

1st Haleema Sadia
School of Engineering
The University of Edinburgh
Edinburgh, Scotland, UK
H.Sadia@sms.ed.ac.uk

2nd Parvez Alam*
School of Engineering
The University of Edinburgh
Edinburgh, Scotland, UK
parvez.alam@ed.ac.uk
*(Corresponding author)

***Abstract*—Superpixel-based segmentation is an important pre-processing step for the simplification of image processing. The subjective nature behind the determination of optimal cluster numbers in segmentation algorithms can result in either under- or over-segmentation burdens, depending on the image type. Insect wings, with their intricate color patterns, pose significant challenges for the accurate capture of color diversity in clustering algorithms, assuming a spherical and isotropic cluster distribution is used. This paper introduces a hybrid approach for color clustering in insect wings, integrating the Simple Linear Iterative Clustering (SLIC) method to generate the initial superpixels, and a DeltaE 2000 function the precisely discriminated merging of superpixels. Color differences between superpixels serve to measure homogeneity during the merging process. The proposed new algorithm demonstrates enhanced segmentation as it overcomes the issue of over-segmentation and under-segmentation, as evidenced by the results derived from the Boundary Recall, Rand index, Under-segmentation Error, and Bhattacharyya distance using ground truth data. The Silhouette score and Dunn Index are also used to quantitatively evaluate the efficacy of our new proposed clustering technique.**

***Index Terms*—Image Processing, Superpixels, SLIC, Color Merging, K-means Clustering**

I. Introduction

Insects such as butterflies, often display intricate color patterns which are present not only from the presence of pigments, but also through structural coloration [1]. Structural coloration results from the interaction of light with nano/microstructural features on the wing surface, which diffracts light to develop iridescence [2]. Color differences induced by changes in lighting and from nano/microscopic structures that create angle-dependent coloration, fall below the threshold of perception in average human eyes [3]. To account for this, specialized image acquisition techniques are commonly used in conjunction with image analysis techniques to enable detailed research on the structural and optical properties of insect wings [2], [4]. In view of the image analysis and computer vision techniques used, clustering algorithms can be seen as playing a key role in decoding and categorizing variable color patterns. Complex color patterns in insect wings coupled with additional challenges like inconsistencies in lighting, high dimensionality, and intricate combinations of hues, pose difficulties for traditional clustering algorithms, particularly in terms of the normalization of coloration [5]. K-means clustering is an unsupervised learning technique widely employed to group pixels by their color similarities, enabling the subsequent segmentation and identification of distinct color regions [6]. Although K-means clustering is unsupervised, the number of clusters, 'K', is still a user-defined parameter and is required due to how the algorithm is designed. The K-means clustering algorithm is computationally efficient, however, over- or under-segmented pixels may result from unsuitable choices for the value of K [7]. Additionally, K-means clustering does not account for spatial relationships in data, leading to potential issues in the effective capturing of spatial coherence. Applying K-means clustering to insect wing colorization thus poses challenges, due to its limitations in capturing spatial dependencies [8].

The application of superpixel segmentation as a pre-processing technique enhances K-means clustering, and rendering the algorithm more robust to variations in intensity, intricacies in color combinations, and irregularities in insect wing color patterns. There are various methods for superpixel segmentation [9], but fundamentally, superpixel segmentation is a computer vision technology, which divides a digital image into distinct identical segments. Each segment, known as a superpixel, constitutes a cluster of pixels, consolidating those that share the highest similarity. Various criteria for homogeneity can be employed to determine the similarity of superpixels, encompassing colors, texture information, and spatial coherence [9]. Generally, an effective superpixel segmentation algorithm yields elevated similarity within a region, while maintaining low inter-region similarity and highlighting significant superpixels. This is as opposed to the retention of the original complexity of thousands of individual pixels [10].

SLIC (Simple Linear Iterative Clustering) is the most widely used superpixel segmentation algorithm in computer vision

We wish to thank the Higher Education Commission (HEC) of Pakistan for funding the PhD scholarship of HS. We are also grateful to Dr. Marcelo Dias for the insightful discussions leading up to the submission of this article.

applications [11], which has already been used for insect wing recognition via color-based segmentation into superpixels for enhanced pattern recognition [12]. It is an effective and enhanced K-means method that requires a predefined number of clusters. SLIC introduces a parameter called "compactness", which influences the shape of the generated superpixels [13]. Additionally, parameter choices including superpixel count and compactness, are known to influence SLIC results. Recent research aims to develop parameter-free approaches to address issues related to providing inappropriate cluster numbers [14].

Image segmentation techniques are categorized in accordance with the image elements employed, specifying distinctions between pixel-wise and superpixel-wise segmentation [15] [16]. The superpixel segmentation groups together neighboring pixels with similar characteristics into perceptually meaningful and homogeneous regions. In contrast to individual pixel-based approaches, the superpixel segmentation approaches reduces both processing time and memory requirements [17]. Nonetheless, a notable limitation of the superpixel-based segmentation approach is in its sensitivity to the initial segmentation quality. The initialization of a high count of superpixels can compromise segmentation efficacy, especially in images with low-contrast edges, shadows, and which contain similarities between foreground and background [18].

Superpixel segmentation methods are categorized as either graph-based or gradient ascent [19] [20]. Graph-based methods treat pixels as nodes in a graph, with edges representing pairwise relationships between neighboring pixels. Contrarily, gradient ascent methods delineate boundaries and identify regions of interest within an image. The gradient ascent method is particularly effective in capturing variations in image intensity or texture, making it valuable for superpixel segmentation where a smooth transition between different regions is desirable [21]. Commonly used gradient ascent methods used for image color segmentation include Normalized Cuts, Graph Cuts, Mean Shifts, Quick Shifts, SLIC, and region competition [22].

SLIC requires pre-specifying the number of superpixels, and an improper selection may cause under-segmentation or over-segmentation, impacting the quality of the result [13]. To address the issue of over-segmentation in SLIC, merging algorithms can be employed to combine closely related superpixels to achieve a more balanced and meaningful segmentation. Three critical considerations in the workflow of region merging include determining the starting point for merging regions, the order in which merging takes place, and the criteria defined for merging [23]. Spatial Merging [24], Color Homogeneity Merging [25], Boundary Merging [26], and Texture-Based Merging [27] are the most widely used merging methods used to overcome the issue of over-segmentation. More recent approaches use machine-learning algorithms to combine diverse features and train classifiers, providing probabilities for the merging of regions [28]. Most recent studies show a color quantization algorithm based on a binary splitting formulation of MacQueen's online k-means, which addresses initialization and acceleration issues with significant speed improvement over popular batch k-means algorithms [29] [30].

A high-quality rapid image segmentation framework has been used to identify an optimal threshold using a hierarchical multi-level image segmentation (MLIS) approach [31]. Here, the authors initially segmented textural and noisy images, after which they employed a local information based SLIC approach (LI-SLIC) and over-segmented superpixels were merged based on a probability distribution among superpixels. Importantly, the authors thus show that using this methods enables the automated generation of an appropriate number of superpixels [31]. The generation of an appropriate number of superpixels after initializing SLIC is still an unresolved problem nevertheless, if when the superpixel parameter is adjusted using a quality metrics-based algorithm [32]. A color and texture segmentation algorithm proposed in [33] builds upon SLIC by incorporating region merging to address the limitations of SLIC in considering neighbor information during the segmentation process. Local Binary Patterns (LBP) and color histogram descriptors were employed to compute both texture and color information for individual regions. The normalized cross-correlation was subsequently used to measure the similarity between adjacent superpixels. The author introduced an enhanced boundary adherence SLIC (BSLIC), a modification of SLIC for superpixel generation that uses three main modifications: (1) the hexagonal initialization of cluster centers, (2) the selection of specific edge pixels as centers, and (3) the inclusion of boundary representations in distance measurements. The segmentation quality of BSLIC was assessed using boundary recall and under-segmentation error metrics. Both the hexagonal distribution of cluster centers and the use of exact edge pixels for center initialization, was noted to improve boundary adherence in edge-across superpixels [34].

Recent advancements in neural network-based clustering algorithms have shown promising results in the research field [35]. However, these state-of-the-art methods often demand substantial amounts of labeled data for effective training, facing challenges in training time, interpretability, hyperparameter tuning, and susceptibility to overfitting, unlike traditional clustering algorithms. Despite advancements in neural network-based clustering algorithms, many researchers still prefer combining clustering-based segmentation with neural networks due to its interpretability, computational efficiency, ability to handle unlabeled data, and robustness to noise [36] [37] [38].

In this paper, we propose a hybrid approach employing SLIC and superpixels merging based on color differences as a pre-processing step for K-means clustering, leveraging the benefits of both methods. Our approach consists of three primary phases designed to address over-segmentation caused by the initialization of SLIC with a high count of superpixels. These are: (1) the initialization of superpixels, which produces a set of over-segmented superpixels by SLIC, (2) superpixel merging utilizing the DeltaE 2000 (also known as ΔE_{00})

function to calculate the color similarity between neighboring superpixels (ΔE_{00} is a color difference formula used to quantify the imperceptible difference between two colors [39]), and (3) K-means clustering by choosing the desired number of clusters to group similar superpixels. SLIC provides a spatially coherent initialization that aligns well with the image content and K-means is then applied adaptively within these regions. This allows for more flexible and efficient color clustering, especially in the context of insect wings. ΔE_{00} addresses the issue of over-segmentation in SLIC in this study. The effectiveness of the color approach is evaluated by Boundary Recall (BR), Rand index (RI), Under-segmentation Error (UE), Bhattacharyya distance, Silhouette score, and Dunn index.

II. Methodology

A. Development of a 4-Phase Method

We introduce a novel enhanced K-Means clustering approach incorporated into the AInsectID open-source entomology software developed by using MATLAB R2022a. [40]. AInsectID digitally analyzes high-resolution images of insect wings and bypasses physical capture for color analysis, revealing distinct color patterns in the digital representation. The flow diagram in Figure 1 illustrates the design of our proposed method, which consists of four phases:

1) **Image pre-possessing**: In the first phase, an RGB image is converted into an LAB image. LAB separates brightness (luminance) information (the L channel) from chromatic information (the a and b channels). The LAB color space offers superior handling of color variations across brightness as compared to RGB.
2) **SLIC Clustering**: In the second phase, we apply an SLIC algorithm to segment the image. This enables the production of a substantive number of superpixels, referred to as over-segmentation. It involves the initialization of the center of a superpixel on the grid, S_g, which is achieved by defining the desired number of superpixels denoted, k. N represents the total pixel count in the image as shown in Equation (1).

$$S_g = \sqrt{\frac{N}{k}} \tag{1}$$

A local neighborhood search is conducted to locate pixels that are close in both color and spatial distance to each superpixel center. Once found, each pixel is then assigned to the cluster.

3) **Superpixel Merging**: Merging superpixels in SLIC based on color involves the combining of adjacent superpixels that exhibit similar color characteristics. In the third phase, the pixels are assigned to the nearest center by weighted distance. During this step, each pixel, i, in the superpixel is linked to its closest superpixel neighbor, k, for situations where the search region intersects with i. Exploration for similar pixels involves a search within a region of size $2S_g \times 2S_g$m centered around the center of the superpixel. By constraining the size of the search region, the algorithm is expedited as it reduces the number of distance calculations. The calculation of the weighted distance, D, involves both color and spatial differences as in Equation (2). The normalization of both spatial distance and color distance (N_s, N_c) is performed according to their respective maximum values.

$$D = \sqrt{\left(\frac{S_s}{N_s}\right)^2 + \left(\frac{S_c}{N_c}\right)^2} \tag{2}$$

Equation (3), the color similarity metric, S_c, measures the difference between the color values of two superpixels. It functions as an Euclidean distance in the color space, where p_i and p_j are the values of two pixels in different superpixels, and L, a, and b represent the LAB color channels. The Euclidean distance measures the color similarity between the two superpixels in 5-D color space. If the Euclidean distance is small, the superpixels are considered 'similar' in 5-D color space.

$$S_c(p_i, p_j) = \sqrt{(L_i - L_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2} \tag{3}$$

Moreover, a spatial distance metric, S_s, is defined to measure the spatial proximity of two superpixels, as shown in Equation (4), where (x_i, y_i) and (x_j, y_j) are the pixel coordinates of p_i and p_j, respectively.

$$S_s(p_i, p_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{4}$$

The maximum anticipated spatial distance, N_s, within a given cluster, matches the sampling interval $N_s = S_g$. Conversely, color distances can exhibit significant variations across clusters. Hence, determining the expected maximum color distance (N_c) within a specific cluster is not simple and is often designated as a constant, m. After substituting these premises in Equation (2), we obtain the weighted distance as Equation (5).

$$D = \sqrt{\left(\frac{S_s}{S_g}\right)^2 + \left(\frac{S_c}{m}\right)^2} \tag{5}$$

The superpixel centers undergo iterative updates after the assignment of pixels to their nearest superpixel centers.

4) **K-Means Clustering** After the color merging is complete, each superpixel is represented by its average color and treated as a data point in a feature space. This information is then used as input to the K-Means algorithm, with the desired number of clusters, K. Average colors are calculated by summing the color values of all pixels within the superpixel and dividing by the number of pixels as shown in Equation (6), where L_i, a_i, and b_i are L, A and B channels of all pixels within the superpixel region, N_k represents the total number of pixels in the superpixel, k, and L_k, a_k, b_k indicates average color.

$$L_k, a_k, b_k = \left(\frac{\sum_{i \in k} L_i}{N_k}, \frac{\sum_{i \in k} a_i}{N_k}, \frac{\sum_{i \in k} b_i}{N_k}\right) \tag{6}$$

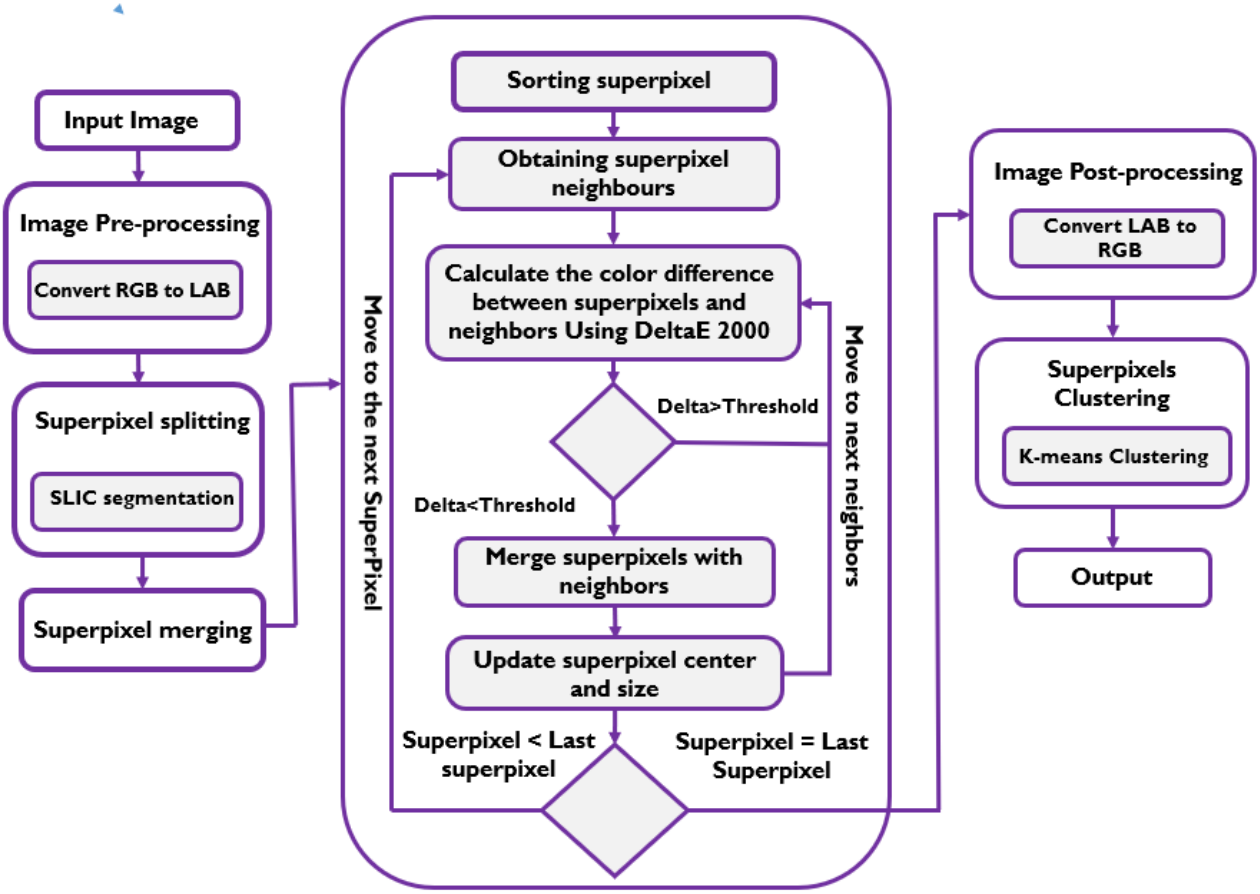

Fig. 1: Flow diagram describing our proposed clustering model as applied in AInsectID for insect wings based on color.

K-Means clustering combines the pixels of superpixels within the same cluster to create larger regions.

B. Color Difference Analysis with DeltaE 2000

In our work, two superpixels were merged based on color differences. The color distance between two superpixels is calculated using the ΔE_{00} function. The merging process involves the selecting of pairs of superpixels with color distances below a threshold, T, after which they are merged into a single superpixel. In color science, Just Noticeable Difference (JND) represents the smallest color difference that an average human eye can detect. A ΔE_{00} value below '1' is often considered as being below the typical JND threshold, meaning that the color difference is so small that it is unlikely to be noticed by the majority of observers under normal viewing conditions [41]. The formula calculates the imperceptible color difference between two superpixels by considering the sensitivity of the human eye to changes in lightness. ($\Delta L'$), chroma ($\Delta C'$), and hue ($\Delta H'$) as shown in Equation (7).

$$\Delta E_{00} = \sqrt{(\Delta L')^2 + (\Delta C')^2 + (\Delta H')^2} \tag{7}$$

After identifying the most suitable matching superpixels, the label of the optimal match is substituted with the label of the k^{th} superpixel. The superpixels with the closest match exhibit a distance below the predefined threshold, T. This procedure was repeated for all pairs of neighboring superpixels until the point in the iterative process where no further superpixels could be merged based on the predefined threshold.

C. Evaluation Metrics

In this study, both qualitative and quantitative assessments were conducted to evaluate the performance of our proposed superpixel segmentation method. Our qualitative approach considers the visual comparison of outputs, while quantitatively, we employ the following six metrics as a means to assessing the differences between methods: Boundary Recall (BR), Rand index (RI), Under-segmentation Error (UE), Bhattacharyya distance (D_B) (each of the preceding being supervised using the ground truth data, and the Silhouette Score and Dunn Index (each of which were used as unsupervised quantitative matrices to evaluate the efficacy of our proposed clustering technique without ground truth). We detected the edges of SLIC and proposed image segmentation using a Canny edge operator [42] in order to effectively compare against ground truth. Each of our supervised and unsupervised quantiative assessment methods are described in more detail below:

- **Boundary Recall (BR)** assesses how effectively the algorithm captures and aligns with the true boundaries between superpixels in an image, specifically focusing on the accuracy of the algorithm in identifying boundaries between color-based superpixels. Boundary recall hence represents the percentage of real edges that closely align with the edges of a superpixel [13]. As in Equation (8), TP refers to the number of pixels that are correctly labeled as belonging to the superpixels in both segmentation and ground truth, while FN refers to the number of pixels that are incorrectly labeled as not belonging to the superpixel in segmentation but belonging to the superpixel in the ground truth. Finally, n represents the total number of pixels within the ground truth superpixel.

$$\text{Recall} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{TP_i}{TP_i + FN_i}\right) \tag{8}$$

  The resulting value of **BR** will range between 0 and 1, where 1 indicates perfect boundary recall, and 0 indicates that no correct boundaries have been identified.
- **Rand Index (RI)** is used to evaluate the similarity between the segmentation results and the ground truth as shown in Equation (9), where FP represents the number of pixels that are incorrectly labeled as belonging to the superpixel in the segmentation but not in the ground truth, and TN represents the number of pixels that are correctly labeled as not belonging to the superpixels in both segmentation and ground truth.

$$RI = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

  The Rand Index ranges from 0 to 1, where 1 indicates a perfect match between segmentation and ground truth, and 0 indicates no agreement [9].
- **Undersegmentation Error (UE)** measures the extent to which the superpixels exceed the boundaries of the ground truth regions. The equation for UE is shown in Equation (10), where S_i is an arbitrary ground truth region, M describes the number of ground truth regions, N is the total number of pixels in the image, p_j refers to any superpixel that intersects with S_i, P_j^{in} shows the section of p_j that overlaps with S_i, and P_j^{out} shows the part of p_j that does not overlap with S_i. The double summation represents the calculation for each ground truth region, S_i, and each superpixel, p_j, that intersects with S_i. The inner summation involves an intersection of a superpixel p_j with the ground truth region (S_i). This minimum pixel count serves as an under-segmentation penalty. A UE value of 0 indicates perfect segmentation with no under-segmentation error, while a value of 1 implies complete failure in adhering to the ground truth boundaries [10].
- **Silhouette Score** is used to measure how well-defined and separated clusters are in a given clustering result. It quantifies the quality of clustering by assessing the cohesion within clusters and the separation between clusters [6]. Equation (11) shows a Silhouette score for a single superpixel, where $a(i)$ refers to the average color distance between a superpixel and other superpixels within the same color cluster, and $b(i)$ defines the smallest average color distance to a superpixel in a different color cluster.

$$\text{Silhouette Score}_i = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{11}$$

  The overall Silhouette score for all superpixels is the average of these individual silhouette scores as shown in Equation (12), where S is the total number of superpixels.

$$\text{Overall Silhouette Score} = \frac{1}{S}\sum_{i=1}^{S}\text{Silhouette Score}_i \tag{12}$$

  Higher Silhouette scores indicate superior superpixel clustering where the superpixels within the same cluster have more color similarities and where clusters are more distinct from one another.
- **Dunn Index** is a clustering validation metric that assesses the compactness of clusters and the separation between different clusters [43]. As shown in Equation (13), $\text{dist}(C_i, C_j)$ is the distance between clusters C_i and C_j (typically the minimum pairwise distance between points in different clusters), and $\text{diam}(C_k)$ is the diameter of a cluster C_k (the maximum pairwise distance between points within the same cluster).

$$\text{Dunn Index} = \frac{\min_{i \neq j}\left(\text{dist}(C_i, C_j)\right)}{\max_k\left(\text{diam}(C_k)\right)} \tag{13}$$

  The Dunn Index is particularly useful for datasets with irregularly shaped clusters, providing insights into the internal structure of clusters. A higher Dunn Index implies that clusters are compact and well-separated.
- **Bhattacharyya Distance (D_B)** is employed to quantify the similarities between the color distributions of segmented images and ground truth. It extracts and compares normalized color histograms from both segmented and ground truth images. The histograms represent the distribution of colors in each image [44]. D_B ranges from 0 to 1, where 0 indicates complete similarity and 1 indicates complete dissimilarity between the two color distributions being compared. In this evaluation, high-resolution microscopic images are used as ground truth. These images capture the precise colors present in the butterfly wing, providing an accurate representation of the intricate color details.

III. Results and Discussion

Both quantitative and qualitative metrics derived from our proposed clustering method are discussed in this section. Figure 2 qualitatively illustrates the visual outcomes for one of the images of size 256×256 from a dataset comprising 200 images of butterfly wings that have a broad variety of colors.

$$UE = \frac{1}{N}\left(\sum_{i=1}^{M}\left(\sum_{p_j:p_j\cap S_i=\emptyset}\min\left(|P_j^{in}|,|P_j^{out}|\right)\right)\right)\times 100\% \tag{10}$$

The ground truth images used in this study were sourced from the Natural History Museum (NHM) butterfly dataset [45], which contains manually annotated images to include details about the wing structure and are available under a CC-BY-4.0 licence. High-resolution microscopic images with high-resolution color details were purchased (standard licence) from the iStock butterfly wing microscopic database [46]. These databases are designed to enable the classification and morphological analysis of butterfly species.

SLIC segmentation was applied with k=500. After segmentation, superpixels were merged based on the proximity of color by ΔE_{00}. Superpixels were merged using various thresholds (ranging from 0 to 1) and compactness values (ranging from 5 to 30). Using these ranges optimal values were identified as m = 10 for compactness and $T = 0.5$ for the threshold.

As observed from the visual outcomes, Figure 2, SLIC produced smaller, and more irregular segments that our proposed clustering model, especially in regions containing subtle variations in color. In contrast, our proposed clustering method utilizes ΔE_{00} to merge segments with perceptually similar colors. This can be understood as important for maintaining high color segmentation quality where the color information plays a vital role. Our method demonstrates improved reductions in fragmenting regions based on minor color differences. By setting a threshold of 0.5, we establish a tolerance level for the color differences. Minor color variations below this threshold are considered similar. Color differences below the threshold that might have led to fragmented regions have now hence been effectively grouped. In this way, the algorithm avoids creating unnecessary regions of fragmentation for subtle color discrepancies, resulting in larger and more homogeneous segments.

To quantitatively support our qualitative visual outputs, we analyzed the segmentation and clustering results as shown in Table I. When we apply boundary recall to butterfly-segmented images and their corresponding ground truth, our method yields high **BR** values when compared against SLIC techniques. This indicates that our color merging approach is adept at identifying boundaries, showcasing its proficiency in preserving and differentiating color clusters. Higher **BR** indicates the proposed segmentation is more coherent and visually meaningful as it reduces the problem of over-segmentation generated by SLIC. Moreover, it preserved color boundaries, allowing for a more accurate representation of color hue in the images. Similarly, in the context of the under-segmentation error, our color merging approach demonstrates low values as compared to SLIC, indicating effectiveness in mitigating under-segmentation issues, as distinguished between various regions based on colors, as it minimizes the merging of distinct colors. This results in a more concise representation of the underlying structures with reduced loss of fine-grained details. In addition, our segmentation shows a high **RI**, which signifies the segmentation method closely aligns with the true grouping of pixels and ascertaining the agreement between proposed and ground truth. Conversely, a low **RI** for traditional SLIC in the case of butterfly images suggests a mismatch between the SLIC and ground truth segmentations, indicating low agreement and potential inaccuracies in the segmentation results as compared to our proposed segmentation.

Moreover, we employed high-resolution microscopic images of butterfly wings as ground truth to quantitatively assess our proposed color clustering method, (cf. Figure 3). We compared between the color distributions obtained through our method and SLIC directly against the high-resolution microscopic images using the Bhattacharyya distance, D_B. Our method results in a distinctly lower D_B when compared against SLIC, indicating that our approach is more effective in preserving the color distribution and similarity between the segmented regions and the ground truth. A lower Bhattacharyya distance signifies a closer match between the color distribution of our method and the ground truth, emphasizing its efficacy in color clustering. Furthermore, we conducted a comprehensive evaluation of our clustering results using both the Silhouette score and the Dunn index. Our outputs reveal a significant improvement in clustering quality as evidenced by an enhancement in both the silhouette score and the Dunn index. This suggests that our method not only enhances the internal cohesion of clusters but also improves their separation, leading to more meaningful and well-defined clustering results. All quantitative matrices tested indicate our proposed method yields enhanced segmentation outcomes.

The merging algorithm can intelligently terminate at convergence, where no further superpixels can be merged based on the defined threshold and the appropriate number of superpixels has been achieved, effectively addressing issues related to under-segmentation. Additionally, We determined less superpixels for an image segmentation task based on our quantitative matrices, while evaluating the performance of the segmentation algorithm across different numbers of superpixels. Our approach achieves boundary adherence while employing an average smaller number of superpixels for images in the insect wings dataset as compared to SLIC. It indicates the effectiveness of our method in producing accurate segmentation results with a reduced computational burden, which is particularly valuable in the context of image processing.

Conclusions and Future Work

In conclusion, our proposed hybrid approach employs SLIC, ΔE_{00} color difference-based merging, and K-means clus-

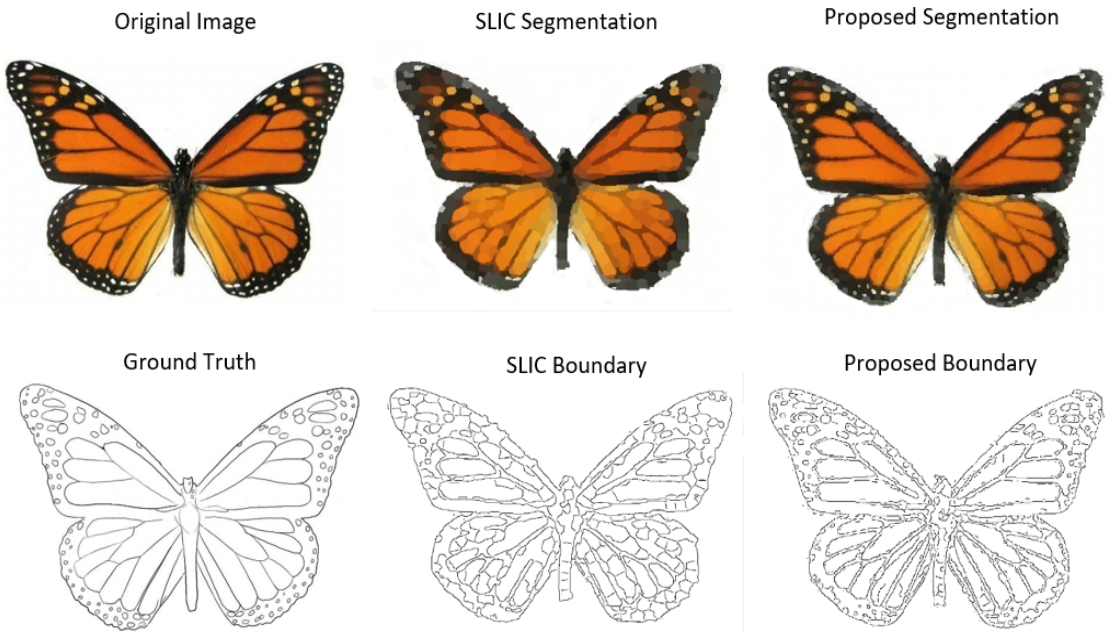

Fig. 2: AInsectID visual segmentation evaluation of SLIC results and those extracted from our proposed model. Original image used with the permission of [45] under a CC-BY-4.0 licence.

TABLE I: Quantitative Evaluation of Proposed Segmentation and Clustering Results

Methods	**Evaluation Metrics**						
	UE	**BR**	**RI**	**Silhouette**	**Dunn**	D_B	**Superpixel Count**
K-Mean	–	–	–	***0.90***	***0.34***		–
SLIC	***0.30***	***0.75***	***0.70***	–	–	***0.7***	***80***
Proposed Segmentation	***0.07***	***0.87***	***0.88***	–	–	***0.3***	***25***
Proposed Clustering	–	–	-	***0.97***	***0.62***	–	–

tering to combines spatial information and perceptual color differences to analyze insect wing colors. Firstly, SLIC effectively considers both color similarity and spatial proximity to create the initial superpixels. Secondly, the ΔE_{00} color difference calculation enhances precision in discriminating subtle color variations, accounting for hue, lightness, and chroma. Benefitting from the strengths of both algorithms, our proposed method achieved more accurate and nuanced clustering of insect wing colors, as compared to solely using K-means and SLIC. Moreover, our proposed approach can automatically determine the number of required superpixels that are considered to be optimal as compared to SLIC, effectively addressing issues related to under-segmentation. This is enabled by leveraging the ΔE_{00} function for enhanced adaptability in the segmentation process. Our proposed method yields enhanced segmentation results compared to previous methods, with reduced computations, which is particularly valuable in the context of insect image processing. Despite the satisfactory performance achieved by the proposed method to automatically decide the number of superpixels by superpixel merging, it still retains a dependency on the optimal value of the threshold. In future work, the merging process could be enhanced by automatically adapting the merging threshold based on the overall image color variance with an aim of converging towards 100% color clustering accuracy.

REFERENCES

[1] R. J. Parchem, M. W. Perry, and N. H. Patel, "Patterns on the insect wing," *Current opinion in genetics & development*, vol. 17, no. 4, pp. 300–308, 2007.

[2] J. Otaki, "The fractal geometry of the nymphalid groundplan: Self-similar configuration of color pattern symmetry systems in butterfly wings. insects 2021, 12, 39," 2021.

[3] L. Zhang, A. Martin, M. W. Perry, K. R. van der Burg, Y. Matsuoka, A. Monteiro, and R. D. Reed, "Genetic basis of melanin pigmentation in butterfly wings," *Genetics*, vol. 205, no. 4, pp. 1537–1550, 2017.

[4] K. Nishida, H. Adachi, M. Moriyama, R. Futahashi, P. E. Hanson, and S. Kondo, "Butterfly wing color made of pigmented liquid," *Cell reports*, vol. 42, no. 8, 2023.

[5] A. K. Davis, B. Herkenhoff, C. Vu, P. A. Barriga, and M. Hassanalian, "How the monarch got its spots: Long-distance migration selects for larger white spots on monarch butterfly wings," *Plos one*, vol. 18, no. 6, p. e0286921, 2023.

[6] J. Yadav and M. Sharma, "A review of k-mean algorithm," *Int. J. Eng. Trends Technol*, vol. 4, no. 7, pp. 2972–2976, 2013.

[7] T. M. Kodinariya, P. R. Makwana *et al.*, "Review on determining number of cluster in k-means clustering," *International Journal*, vol. 1, no. 6, pp. 90–95, 2013.

[8] A. Gupta, R. Singh, V. K. Nassa, R. Bansal, P. Sharma, and K. Koti, "Investigating application and challenges of big data analytics with clustering," in *2021 international conference on advancements in electrical, electronics, communication, computing and automation (ICAECA)*. IEEE, 2021.

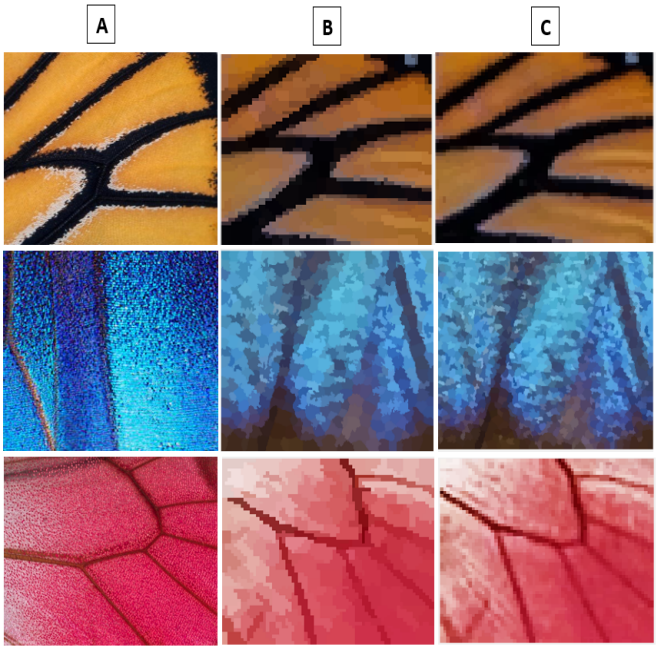

Fig. 3: Visual evaluation of color distribution of segmentation outputs: (A) high-resolution microscopic images, (B) SLIC segmentation, and (C) our proposed segmentation. Images purchased under a standard licence from the butterfly wing microscopic collection available in iStock, [46]

.

[9] B. V. Kumar, "An extensive survey on superpixel segmentation: A research perspective," *Archives of Computational Methods in Engineering*, pp. 1–19, 2023.

[10] A. Ibrahim and E.-S. M. El-kenawy, "Image segmentation methods based on superpixel techniques: A survey," *Journal of Computer Science and Information Systems*, vol. 15, no. 3, pp. 1–11, 2020.

[11] S. Vasantha, S. Srujana, G. Swetha, and V. Manasa, "Animated and oil painting image generation from real world images using super-pixel based slic algorithm," in *2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*. IEEE, 2023.

[12] S. D. Mahalakshmi and K. Vijayalakshmi, "Retracted article: Agro suraksha: pest and disease detection for corn field using image analysis," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 7375–7389, 2021.

[13] S. Di, M. Liao, Y. Zhao, Y. Li, and Y. Zeng, "Image superpixel segmentation based on hierarchical multi-level li-slic," *Optics & Laser Technology*, vol. 135, p. 106703, 2021.

[14] S. Subudhi, R. N. Patro, P. K. Biswal, and F. Dell'Acqua, "A survey on superpixel segmentation as a preprocessing step in hyperspectral image analysis," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 5015–5035, 2021.

[15] B. Sasmal and K. G. Dhal, "A survey on the utilization of superpixel image for clustering based image segmentation," *Multimedia Tools and Applications*, pp. 1–63, 2023.

[16] Y. Yu, C. Wang, Q. Fu, R. Kou, F. Huang, B. Yang, T. Yang, and M. Gao, "Techniques and challenges of image segmentation: A review," *Electronics*, vol. 12, no. 5, p. 1199, 2023.

[17] B. A. Taha, Q. Al-Jubouri, Y. Al Mashhadany, M. H. H. Mokhtar, M. S. D. B. Zan, A. A. A. Bakar, and N. Arsad, "Density estimation of sars-cov2 spike proteins using super pixels segmentation technique," *Applied soft computing*, vol. 138, p. 110210, 2023.

[18] B. V. Kumar, "An extensive survey on superpixel segmentation: A research perspective," *Archives of Computational Methods in Engineering*, pp. 1–19, 2023.

[19] A. Fejjari, K. S. Ettabaa, and O. Korbaa, "Improved superpixels generation algorithm for qualified graph-based technique," *INTERNATIONAL ARAB JOURNAL OF INFORMATION TECHNOLOGY*, vol. 19, no. 6, pp. 949–955, 2022.

[20] A. Mansouri, J.-C. Créput, and W.-B. Qiao, "Generic parallel data structures and algorithms to gpu superpixel image segmentation," *Displays*, vol. 74, p. 102275, 2022.

[21] B. Ji, X. Hu, F. Ding, Y. Ji, and H. Gao, "An effective color image segmentation approach using superpixel-neutrosophic c-means clustering and gradient-structural similarity," *Optik*, vol. 260, p. 169039, 2022.

[22] A. Rafi, Z. Khan, F. Aslam, S. Jawed, A. Shafique, and H. Ali, "A review: Recent automatic algorithms for the segmentation of brain tumor mri," *AI and IoT for Sustainable Development in Emerging Countries: Challenges and Opportunities*, pp. 505–522, 2022.

[23] E. Güngör and A. Özmen, "Coarse segmentation with gdd clustering using color and spatial data," *IEEE Access*, vol. 8, pp. 144 880–144 891, 2020.

[24] F. Ma, F. Zhang, D. Xiang, Q. Yin, and Y. Zhou, "Fast task-specific region merging for sar image segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–16, 2022.

[25] S. Habib, I. Khan, S. Aladhadh, M. Islam, and S. Khan, "External features-based approach to date grading and analysis with image processing," *Emerg. Sci. J*, vol. 6, no. 4, pp. 694–704, 2022.

[26] H. Yu, H. Jiang, Z. Liu, Y. Sun, S. Zhou, and Q. Gou, "Sar image segmentation by merging multiple feature regions," in *2022 3rd International Conference on Geology, Mapping and Remote Sensing (ICGMRS)*. IEEE, 2022.

[27] H. S. Lee and S. In Cho, "Spatial color histogram-based image segmentation using texture-aware region merging," *Multimedia Tools and Applications*, vol. 81, no. 17, pp. 24 573–24 600, 2022.

[28] J. Yu, F. Wellmann, S. Virgo, M. von Domarus, M. Jiang, J. Schmatz, and B. Leibe, "Superpixel segmentations for thin sections: Evaluation of methods to enable the generation of machine learning training data sets," *Computers & Geosciences*, vol. 170, p. 105232, 2023.

[29] A. Abernathy and M. E. Celebi, "The incremental online k-means clustering algorithm and its application to color quantization," *Expert Systems with Applications*, vol. 207, p. 117927, 2022.

[30] S. Thompson, M. E. Celebi, and K. H. Buck, "Fast color quantization using macqueen's k-means algorithm," *Journal of Real-Time Image Processing*, vol. 17, no. 5, pp. 1609–1624, 2020.

[31] Z. Luo, W. Yang, Y. Yuan, R. Gou, and X. Li, "Semantic segmentation of agricultural images: a survey," *Information Processing in Agriculture*, 2023.

[32] K. Sabaneh and M. Sabha, "Improving slic superpixel by color difference-based region merging," *Multimedia Tools and Applications*, pp. 1–19, 2022.

[33] Y. Qian, Y. Xue, and T. Wang, "Deep interactive image segmentation based on region and boundary-click guidance," *Journal of Visual Communication and Image Representation*, vol. 92, p. 103797, 2023.

[34] H. Wang, X. Peng, X. Xiao, and Y. Liu, "Bslic: Slic superpixels based on boundary term," *Symmetry*, vol. 9, no. 3, p. 31, 2017.

[35] M. R. Karim, O. Beyan, A. Zappa, I. G. Costa, D. Rebholz-Schuhmann, M. Cochez, and S. Decker, "Deep learning-based clustering approaches for bioinformatics," *Briefings in bioinformatics*, vol. 22, no. 1, pp. 393–415, 2021.

[36] J. Kauffmann, M. Esders, L. Ruff, G. Montavon, W. Samek, and K.-R. Müller, "From clustering to cluster explanations via neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[37] A. K. Sahoo, P. Parida, and K. Muralibabu, "Hybrid deep neural network with clustering algorithms for effective gliomas segmentation," *International Journal of System Assurance Engineering and Management*, vol. 15, no. 3, pp. 964–980, 2024.

[38] F. Liu, T. Dong, Q. Liu, Y. Liu, and S. Li, "Combining fuzzy clustering and improved long short-term memory neural networks for short-term load forecasting," *Electric Power Systems Research*, vol. 226, p. 109967, 2024.

[39] N. Yuan, X. Zhao, B. Sun, W. Han, J. Tan, T. Duan, and X. Gao, "Low-light image enhancement by combining transformer and convolutional neural network," *Mathematics*, vol. 11, no. 7, p. 1657, 2023.

[40] H. Sadia and P. Alam, "AInsectID - a free-to-use species identification, image analysis and color mapping software," 2023. [Online]. Available: https://doi.org/10.7488/ds/7553

[41] L. A. Moreira, M. Watsa, G. Erkenswick, J. P. Higham, and A. D. Melin, "Evaluating genital skin color as a putative sexual signal in wild

saddleback (leontocebus weddelli) and emperor (saguinus imperator) tamarins," *American Journal of Primatology*, vol. 85, no. 2, p. e23456, 2023.

[42] L. Wang, X. Ma, and H. Wang, "Hybrid image edge detection algorithm based on fractional differential and canny operator," in *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2. IEEE, 2018, pp. 210–213.

[43] M. Misuraca, M. Spano, and S. Balbi, "Bms: An improved dunn index for document clustering validation," *Communications in statistics-theory and methods*, vol. 48, no. 20, pp. 5036–5049, 2019.

[44] M. Tejonidhi, B. Nanjesh, J. G. Math, and A. G. D'sa, "Plant disease analysis using histogram matching based on bhattacharya's distance calculation," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. IEEE, 2016, pp. 1546–1549.

[45] Natural History Museum, "Collection specimens dataset," https://data.nhm.ac.uk/dataset/collection-specimens/resource/05ff2255-c38a-40c9-b657-4ccb55ab2feb, accessed: January 19, 2024, (Creative Commons Licence - CC-BY-4.0).

[46] "Butterfly wing microscopic," https://www.istockphoto.com/photos/butterfly-wing-microscopic, date purchased: January 20, 2024.