






# A Study of Lab Color Space and Its Visualization

Ida Ayu Putu Febri Imawati<sup>1</sup>, Made Sudarma<sup>2</sup>  
I Ketut Gede Darma Putra<sup>3</sup>, I Putu Agung Bayupati<sup>4</sup>

<sup>1</sup> Study Program of Doctoral Engineering Science Faculty of Engineering Udayana University, Denpasar – Indonesia

<sup>2</sup> Department of Electrical Engineering Udayana University, Denpasar – Indonesia

<sup>3,4</sup> Department of Information Technology Udayana University, Denpasar – Indonesia  
\*imawati.2291011018@student.unud.ac.id

**Abstract.** With the increasing need for digital images in everyday life, images are collected through various devices such as digital cameras, cell phone cameras, and scanners. This image data will be further processed, one of which is to segment objects from the background. The technique that can be used is segmentation using the LAB color space. This technique is done by converting the image color space into LAB color space so that the object or foreground can be separated from the background. This research uses 20 random images from 3 sources: The Oxford-IIIT Pet dataset, Github Real Python material, and DeepLontar dataset. The experimental results show that The Oxford-IIIT Pet dataset and Github Real Python material have a more extended range of minimum-maximum values of L, a\*, and b\* components compared to DeepLontar dataset. This extended minimum-maximum value range causes the object images in The Oxford-IIIT Pet dataset and Github Real Python materials to be more visually visible (segmented) than in the DeepLontar dataset.

**Keywords:** Color space, Cielab, Color space segmentation, Image processing, Preprocessing.

## 1 Introduction

Currently, the need to use digital images in everyday life is increasing. These digital images are recorded through various devices such as digital cameras, cell phone cameras, and scanners. The need for digital images is widely used to make copies of documents stored and back up physical files into digital documents.

Digital images that have been obtained can also be further processed, such as in the process of recognition or detection of characters [1], [2], hand geometry detection [3], and face recognition for the authentication process [4]. However, the resulting digital images sometimes have relatively similar foregrounds and backgrounds, so they experience difficulties when further processing [5]. Therefore, object segmentation is needed to separate the object from the background. Segmentation is a process to separate an object from the background so that the object can be processed for further purposes [3], [4], [5].

© The Author(s) 2024

K. Dharmawan and N. A. Sanjaya ER (eds.), *Proceedings of the First International Conference on Applied Mathematics, Statistics, and Computing (ICAMSAC 2023)*, Advances in Computer Science Research 110, [https://doi.org/10.2991/978-94-6463-413-6\\_3](https://doi.org/10.2991/978-94-6463-413-6_3)

Digital images that have been obtained can also be further processed, such as in the process of recognition or detection of characters [1], [2], hand geometry detection [3], and face recognition for the authentication process [4]. However, the resulting digital images sometimes have relatively similar foregrounds and backgrounds, so they experience difficulties when further processing[5]. Therefore, object segmentation is needed to separate the object from the background. Segmentation is a process to separate an object from the background so that the object can be processed for further purposes [3], [4], [5].

## 2 Literatures review

### 2.1 RGB Color Space

By using the characteristics of color and light intensity, we can recognize images. Colorspace is used as one of the techniques in digital image processing. RGB color space (Red, Green, Blue) is a color space that is based on how the human eye works [6]. The human eye has two sensors on retina sensors: rod and cone cells. In the processing of digital image analysis, color space RGB is preferred because it does not require conversion to another color space. The color produced follows the vision representation of the human eye.

### 2.2 Lab Color Space

This color space was defined by the CIE in 1976 to communicate color widely, which is widely used in industry for color control and management. CIELAB is a three-dimensional color space model dimensions. In this color space, a slice is taken from components  $a^*$  and  $b^*$ , and from the slices of components  $a^*$  and  $b^*$  obtained  $a^* b^*$  chromaticity diagram with the meaning of each dimension formed.

CIE  $L^*$  magnitude is used to show the description of color brightness. The CIE  $a^*$  dimension is used to indicate the description of the color type green - red color type. The CIE  $b^*$  dimension is used for the blue-yellow color type [6], [7]. The RGB to CIELAB color space conversion calculation can be presented as in equation 1 to equation 4 obtained from the OpenCV library[8].

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \leftarrow \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

Description:

R = Red color values

G = Green color values

B = Blue color values

After converting to XYZ, the CIELAB color space can be calculated using Equation (2).

$$\begin{aligned}
L &\leftarrow 116 * Y^{1/3} - 16 \text{ For } Y > 0.008856 \\
L &\leftarrow 903.3 * Y \text{ For } Y \leq 0.008856 \\
a^* &\leftarrow 500(f(X) - f(Y)) \\
b^* &\leftarrow 200(f(Y) - f(Z))
\end{aligned} \tag{2}$$

The reference values of  $X_n$ ,  $Y_n$ , and  $Z_n$  can be seen in Equation (3).

$$\begin{bmatrix} X_n \\ Y_n \\ Z_n \end{bmatrix} = \begin{bmatrix} 0.950456 \\ 1.0 \\ 1.088754 \end{bmatrix} \tag{3}$$

The transformation function equation  $f(t)$  can be implemented as in in Equation (4).

$$f(t) = \begin{cases} t^{1/3} & \text{for } t > 0.008856 \\ 7.787t + \frac{16}{116} & \text{for } t \leq 0.008856 \end{cases} \tag{4}$$

**The advantages of CIELab.** This research aims to implement CIELab on 20 images to represent color features. Color features are one of the features that separate image objects from their background (color space segmentation). This color space segmentation can be done on facial images, batik, and copper and papyrus inscriptions. Inscriptions are unique because there is a low color difference between the letters (objects) and the media for writing them.

Many studies related to CIELab have been conducted. Fawaz et al. [4], the CIELab model is used because it can express the colors seen by the human eye, and also, this model compensates for the inequality of the color distribution of the RGB color model because the RGB model has too many transition colors between blue and green. The CIELab model was chosen to identify the characteristics of a batik cloth [7], [9]. Identifying the characteristics of batik cloth is necessary because the development of batik cloth patterns resulting from manual coloring or printed motifs is easy to imitate and reproduce. In addition, the CIELAB model was chosen because of the homogeneous space for visual perception.

The CIELab model is also implemented on inscriptions. Based on Sudarma's papyrus lontar research results [5], CIELab color space can identify and separate Balinese script colors from their background colors. Using the  $L^*$  value of the CIELab, experimental results show the brightness of the color as a little differentiator. Rasmana's research [10] measured the color difference of carved letters with copper inscription plates. The measurement was done with the CIELAB model. The results of this research state that the color feature is unsuitable for recognizing the carved letters on the inscription because the color difference between the carved letters and the plate is tiny. The measurement results of the color difference in the  $a^*$  and  $b^*$  layers are small. The big difference is in the  $L^*$  layer. The  $L^*$  layer is the layer that represents the color intensity of the image. However, this research uses the  $L^*$  layer combined with texture features for the following segmentation process. Previous studies representing the CIELab color space

were limited to only one specific object or image. The authors conducted CIELab color space experiments on three data sources in this study. This research contributes to illustrating that CIELab color space can be one way of extracting image features with different image characteristics.

### 3 Method

#### 3.1 Preprocessing and Generating Data

This research uses 20 images. The images were obtained from several sources: The Oxford-IIIT Pet Dataset [11], Github Real Python material [12], and the DeepLontar dataset [13]. The author preprocessed the image first on the DeepLontar dataset ground truth data, namely the image with the name 8a.jpg, as in Fig 1. The preprocessing was done because the image dimensions were too large at 1500 x 300 pixels. The image from the DeepLontar dataset is cropped randomly with varying dimensions: 1) the author cropped per character as many as five characters to see the visualization results per character, 2) the author also cropped randomly (5 times) for a collection of several characters or free text along the dimension of 1500 x 300 pixels in the image 8a.jpg.

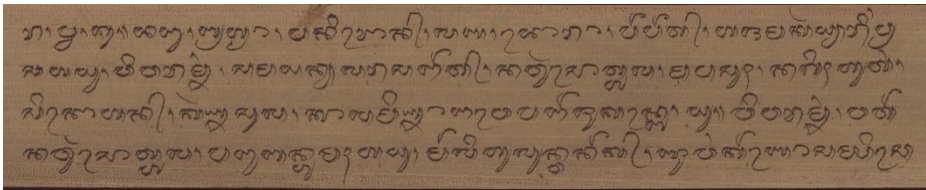


Fig. 1. An image from DeepLontar Dataset namely 8a.jpg.

Second, for the dataset from The Oxford-IIIT Pet Dataset, the author took five random image samples (Bombay\_220.jpg, Bengal\_197.jpg, Ragdoll\_27.jpg, Sphynx\_142.jpg and Russian\_Blue\_187.jpg) and from Github Real Python material as well as five images (nemo1.jpg, nemo2.jpg, nemo3.jpg, nemo4.jpg, nemo5.jpg), as in Fig 3. The image generation process, as in Fig 2.

#### 3.2 Implemented Algorithm

This algorithm is modified from the GitHub source [14] and applied, as shown in Figure 3.

1. Read image data and load example images.
2. Read the five images, then save them into a list.
3. Convert each of the images into numpy array.
4. Extract the first few example images. RGB color images consist of three layers: a red layer, a green layer, and a blue layer. Each layer in a color image has a value

from 0 - 255. In this layer, the image mean has no color if the value is 0, and the pixel is black if the value is 0 for all color channels.

5. Convert RGB to LAB. Conversion from the sRGB color space (IEC 61966-2-1:1999) to the CIE Lab colorspace under the given illuminant and observer. The range of the dimensions for RGB and LAB in skimage.color.rgb2lab and lab2rgb are:  $rgb\_lab:[0,1] \times [0,1] \times [0,1] \rightarrow [0,100] \times [-128,128] \times [-128,128]$  and  $lab\_rgb:[0,100] \times [-128,128] \times [-128,128] \rightarrow [0,1] \times [0,1] \times [0,1]$ .
6. Inverse the process (LAB to RGB), lab2rgb has to have a dimension (-,-, 3).
7. Check if the RGB  $\rightarrow$  LAB  $\rightarrow$  RGB worked by plotting.
8. Check if the 0th dimension of the LAB image is showing the brightness. Finally, each dimension of the LAB image can be visualized.

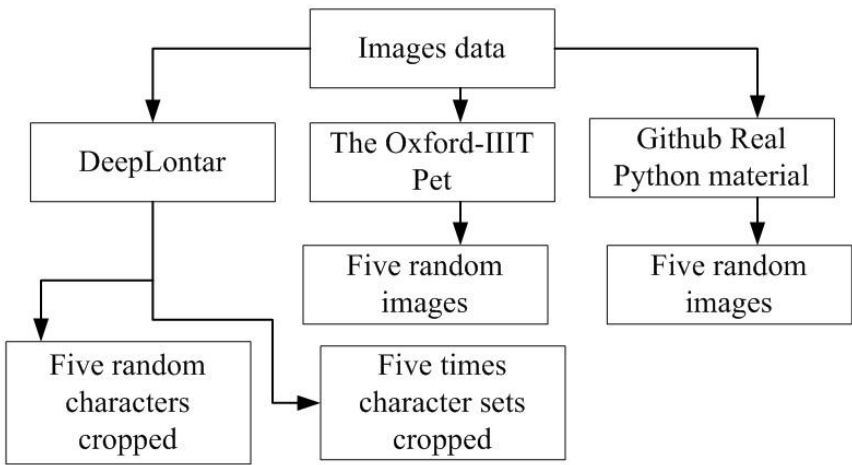












Fig. 2. Image generation process

Table 1. Image data

Data source name	Images				
The Oxford-IIIT					
	Bombay_220.jpg	Bengal_197.jpg	Ragdoll_27.jpg	Sphynx_142.jpg	Russian_Blue_187.jpg
Github Re-alpython material					
	nemo1.jpg	nemo2.jpg	nemo3.jpg	nemo4.jpg	nemo5.jpg

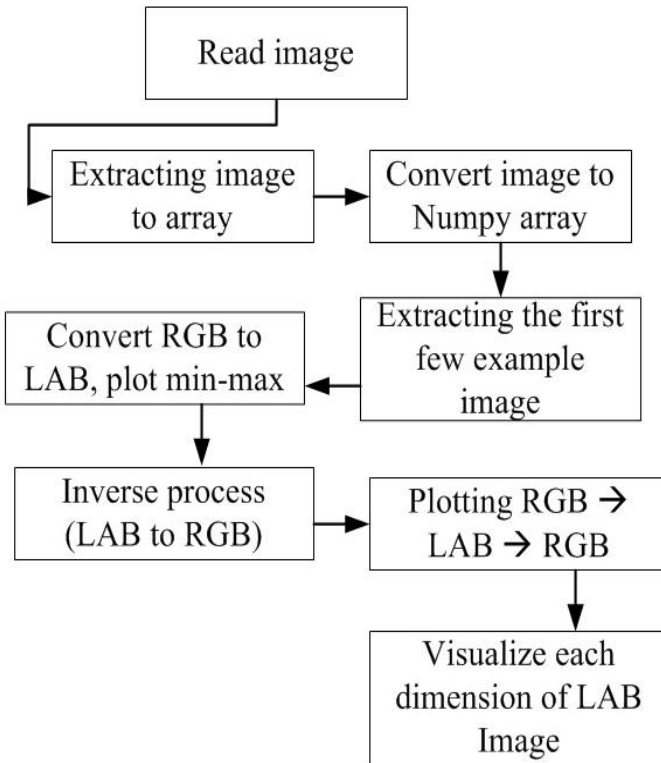
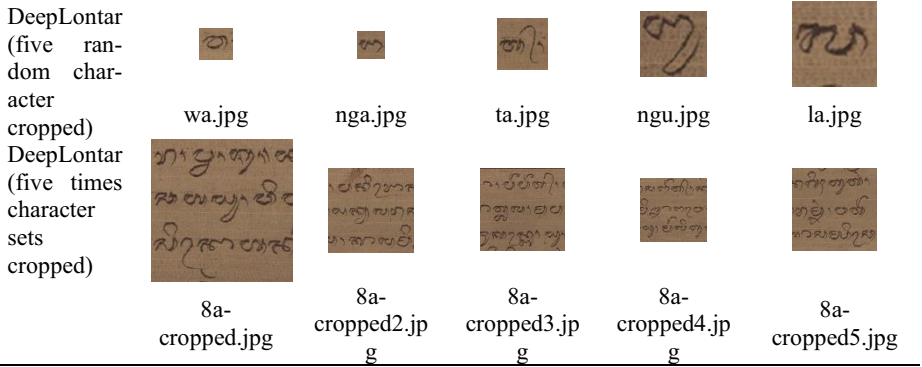


Fig. 3. Implemented algorithm

## 4 Result and Discussion

The results of the experiments in this study are in the form of values or values of each component, namely the minimum-maximum value in RGB color space and the minimum-maximum value in LAB color space, as shown in Table 2 and Table 3. Based on Tables 2 and Table 3, The Oxford-IIIT Pet dataset and Github Real Python have the same minimum to a maximum range of R, G, and B values, namely 0 to 255. Similarly, the same L (Lightness) value is a minimum of 0 to a maximum of 100. It implicitly states that the images on The Oxford-IIIT Pet dataset and Github Real Python have a more extended range of values in RGB color space and LAB color space. For the DeepLontar dataset, both cropped per character and randomly cropped have a shorter range of values in both color spaces.

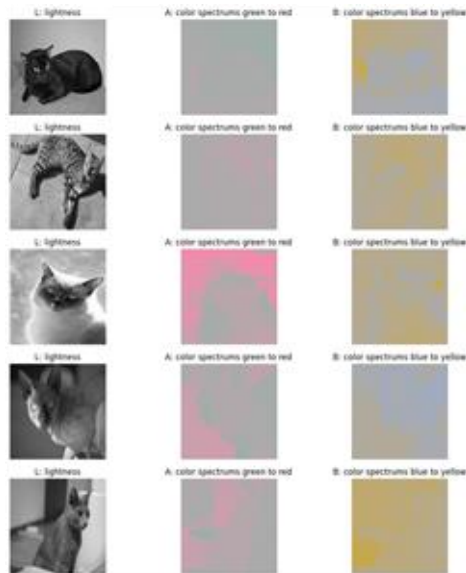
Values with a more extended range will impact the visualization of the L, a\*, and b\* components. Visually, we can see the object (foreground) separated from the background, as in Figure 4 and Figure 5. In Figure 5, the image obtained from GitHub Real Python has a range that tends to be significantly prolonged so that we can visually see the segmented fish object from the background. Figure 6 and Figure 7 are data taken from lontar or papyrus. They have shorter range of minimum to maximum values of the color space components (both in RGB and LAB). The LontarDeep dataset has been dealt with by cropping per character (Figure 6), which makes the characters more evident so that in the b\* component (color spectrum blue to yellow), visually, we can still see the segmented object from the background.

**Table 2.** The result of min max value of RGB

Data	R		G		B	
	min	max	min	max	min	max
The Oxford-IIIT Pet	0.0000	255.00000	0.0000	255.00000	0.0000	255.00000
Github Real Python	0.0000	255.00000	0.0000	255.00000	0.0000	255.00000
DeepLontar (five random character cropped)	42.0000	169.0000	26.0000	143.0000	14.0000	112.0000
DeepLontar (five times character sets cropped)	25.0000	174.0000	10.0000	144.0000	13.0000	116.0000

**Table 3.** The Result of min max value of LAB

Data	L		a*		b*	
	min	max	min	max	min	max
The Oxford-IIIT Pet Github	0.0000	100.0000	-29.3345	66.2213	-59.8455	68.8716
Real Python	0.0000	100.0000	-31.5718	68.9017	-98.6082	81.6538
DeepLontar (five random character cropped)	11.3153	60.7135	-1.5736	14.1609	-0.7475	27.7050
DeepLontar (five times character sets cropped)	4.2075	61.2767	-1.4154	22.2171	-7.3443	27.5035



**Fig. 4.** Lab visualizations of The Oxford-IIIT Pet data



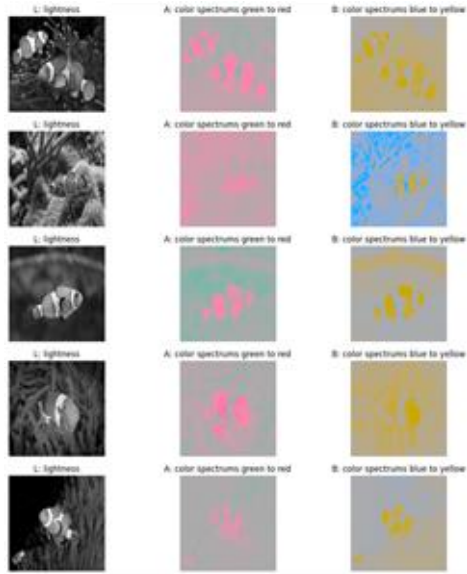


Fig. 5. . Lab Visualizations of Github real Python data

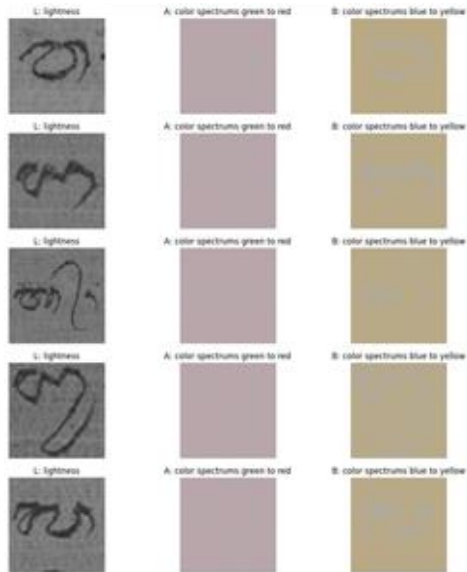
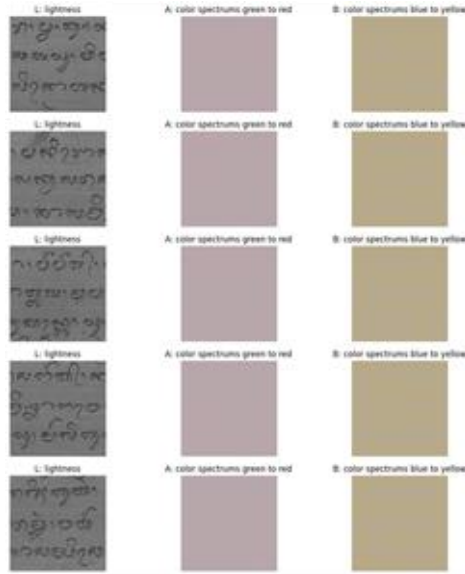


Fig. 6. Lab visualizations of DeepLontar data (five random character cropped)



**Fig. 7.** Lab visualizations of DeepLontar data (five times character sets cropped)

As stated by Sudarma[5] and Rasmana [10] in their research using inscription objects, inscriptions tend to have slight color differences between the object and the media on which it is written. Of the three layers in the CIELab color model, the most significant difference is in the  $L^*$  layer. The  $L^*$  layer is the layer that represents the color intensity of the image. The results of the two studies align with the results of the range of min-max  $L^*$ ,  $a^*$ , and  $b^*$  values in Table 3. In DeepLontar (five random characters cropped), the  $L^*$  layer values range from 11.3153 to a maximum of 60.7135,  $a^*$  layer values range from -1.5736 to 14.1609, and  $b^*$  layer values -0.7475 to 27.7050. In DeepLontar (five times character sets cropped), layer  $L^*$  values range from 4.2075 to 61.2767, layer  $a^*$  values from -1.4154 to 22.2171, and layer  $b^*$  values range from -7.3443 to 27.5035. Layer  $L^*$  thus has a more extended range of values than the other two layers. Therefore, CIELab can be used to extract color features in images. However, suppose the characteristics of the object to be separated or segmented from an image have the same characteristics as the inscription. In that case, it must be combined with other feature extraction to get better segmentation results[10].

## 5 Conclusions

The longer the range of minimum to maximum values of the R, G, B, or L,  $a^*$ ,  $b^*$  components in each color space, the more convenient we can see the segmented object from the background. On the other hand, the shorter the range of minimum to maximum values of the R, G, B, or L,  $a^*$ ,  $b^*$  components in each color space, the more difficult it is to segment the object from the background. This research uses input images with various dimensions, so for further research, it is necessary to standardize the dimensions of the input image and the opportunity to combine this CIELab color feature extraction

with other image feature extraction methods such as texture, pattern, shape, or edge features.

## References

1. Anggraeni DT, Wibawa C. Perbaikan Citra Tanda Tangan Digital Menggunakan Metode Otsu Thresholding dan Sauvola. *J Ilm Matrik* 2023;25:28–34. <https://doi.org/10.33557/jurnalatrik.v25i1.2324>.
2. Anggraeni DT. Perbaikan Citra Dokumen Hasil Pindai Menggunakan Metode Simple, Adaptive-Gaussian, dan Otsu Binarization Thresholding. *Expert J Manaj Sist Inf Dan Teknol* 2021;11:71. <https://doi.org/10.36448/expert.v11i2.2170>.
3. Putra D. Binerisasi Citra Tangan dengan Metode Osu. *Maj Ilm Teknol Elektro* 2004;3:11–3. <https://doi.org/10.24843/mite.2015.v14i02p09>.
4. Fawaz A, Hakimah M, Kurniawan M. Segmentasi Citra Wajah Dengan Menggunakan Metode K-Means –  $L^*a^*b$ . *Pros Semin Nas Sains Dan Teknol Terap* 2021;9:493–9.
5. Sudarma M. Identifying of the Cielab Space Color for the Balinese Papyrus Characters. *TELKOMNIKA Indones J Electr Eng* 2015;13. <https://doi.org/10.11591/telkomnika.v13i2.7086>.
6. Fatham M, Akbar M, Fitriyah H, Akbar SR. Analisis Color Space untuk Spesifikasi Perancangan Perangkat Lunak pada Embedded System Deteksi Penyakit Busuk Selada 2023;7:2186–93.
7. Sinaga AS. SEGMENTASI RUANG WARNA  $L^*a^*b$ . *J Mantik Penusa* 2019;3:43–6.
8. OpenCV.org. OpenCV Color Conversion. *OpenCVOrg* 2023:2–5. [https://docs.opencv.org/3.4/de/d25/imgproc\\_color\\_conversions.html](https://docs.opencv.org/3.4/de/d25/imgproc_color_conversions.html) (accessed September 15, 2023).
9. Sinaga ASR. Color-based Segmentation of Batik Using the  $L^*a^*b$  Color Space. *Sinkron* 2019;3:175. <https://doi.org/10.33395/sinkron.v3i2.10102>.
10. Rasmana ST. Letter Segmentation of the Ancient Copper Inscriptions Using Texture-Based. *Institut Teknologi Sepuluh Nopember*, 2017.
11. Parkhi OM, Vedaldi A, Zisserman A, Jawahar C V. Cats and dogs. *Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit* 2012:3498–505. <https://doi.org/10.1109/CVPR.2012.6248092>.
12. Stone R. Image Segmentation Using Color Spaces in OpenCV + Python – Real Python. *GithubCom/Realpython/Materials* 2018. <https://realpython.com/python-opencv-color-spaces/>.
13. Siahaan D, Sutramiani NP, Suciati N, Duija IN, Darma IWAS. DeepLontar dataset for handwritten Balinese character detection and syllable recognition on Lontar manuscript. *Sci Data* 2022;9:1–7. <https://doi.org/10.1038/s41597-022-01867-5>.
14. Kondo Y. Color-space-defenitions-in-python-RGB-and-LAB. *FairyOnIceGithubIo* 2020. <https://github.com/FairyOnIce/FairyOnIce.github.io>.

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

