Evaluating the attributes of remote sensing image pixels for fast k-means clustering

Ali SAĞLAM*, Nurdan AKHAN BAYKAN
Department of Computer Engineering, Faculty of Engineering and Natural Sciences,
Konya Technical University, Konya, Turkey

Abstract: Clustering process is an important stage for many data mining applications. In this process, data elements are grouped according to their similarities. One of the most known clustering algorithms is the k-means algorithm. The algorithm initially requires the number of clusters as a parameter and runs iteratively. Many remote sensing image processing applications usually need the clustering stage like many image processing applications. Remote sensing images provide more information about the environments with the development of the multispectral sensor and laser technologies. In the dataset used in this paper, the infrared (IR) and the digital surface maps (DSM) are also supplied besides the red (R), the green (G), and the blue (B) color values of the pixels. However, remote sensing images come with very large sizes (6000 × 6000 pixels for each image in the dataset used). Clustering these large-size images using their multiattributes consumes too much time if it is used directly. In the literature, some studies are available to accelerate the k-means algorithm. One of them is the normalized distance value (NDV)-based fast k-means algorithm that benefits from the speed of the histogram-based approach and uses the multiattributes of the pixels. In this paper, we evaluated the effects of these attributes on the correctness of the clustering process with different color space transformations and distance measurements. We give the success results as peak signal-to-noise ratio and structural similarity index values using two different types of reference data (the source images and the ground-truth images) separately. Finally, we give the results based on accuracy measurement for evaluating both the success of the clustering outputs and the reliability of the NDV-based measurement methods presented in this paper.

Key words: Remote sensing images, clustering, k-means, color transformation, distance norms

1. Introduction

In machine vision systems, extracting meaningful information from digital images obtained by spectral sensors is an important and main occupation for the researchers. For this purpose, the machine vision applications make use of the data mining methods. Clustering methods that collect similar data elements into one group and separate dissimilar ones into different groups are some of the most used data mining methods and the k-means clustering method is a well-known clustering method [1].

In the k-means clustering method, all the data elements are assigned to a cluster in each iteration, according to the closeness to the cluster centers. At first, the number of clusters is given as a parameter and the centers of the cluster are positioned (randomly in this work). In each of the iterations, each element is assigned to a cluster whose center is nearest to the element and the cluster centers are repositioned by calculating the

---

*Correspondence: alisaglam666@gmail.com

4188

central position of the elements in the same cluster. In the next iteration, all the elements are assigned to one of the clusters again according to the new cluster centers. When it reaches the specified number of iteration or there is no change in the cluster centers, the algorithm ends [1].

The k-means algorithm runs inconvenient time if it is based on the histogram-based approaches [2, 3]. A histogram shows the number of elements such that each has the same attributes. When an element is assigned to a cluster, the elements that have the same attribute value are assigned to the same cluster. This process accelerates the clustering algorithm by preventing proximity calculation for all elements. To express the data with a useful histogram, each data element must have a single property value and the value must be discrete in a specified range; for example the integers in the 8-bit pixel value interval range from 0 to 255 [4].

Remote sensing images have usually very large sizes and include an enormous number of pixels. For this reason, to cluster the remote sensing images in an acceptable time, a fast approach such as the histogram-based k-means method is needed. A remote sensing image pixel has generally multiattribute values, such as RGB color values (R, red; G, green; and B, blue), the infrared (IR) and the digital surface maps (DSM) [5, 6]. Some of the simple ways to express these pixels with histogram are the procedure of converting the color value vectors to the gray values or the procedure of using one attribute band. Baykan and Saglam proposed a histogram-based method that makes use of all attribute values of image pixels and applied it on remote sensing images [4]. The results show that the method gives more accurate clustering results compared to the gray-level histogram. According to the method, an attribute vector is determined such as mean attribute vector of the image to be clustered and then, the distance values, for example, Euclidean distance, are calculated from the elements to the center, separately. The distance values are normalized in a specified range and discretized. These values are named normalized distance values (NDVs) in the study.

In this paper, we evaluate the effects of the attributes of the remote sensing image pixels and the distance norms (distance measurements) on the k-means clustering performance using their different combinations and color transformations for NDVs. To evaluate the clustering performances, firstly, we use peak signal-to-noise rate (PSNR) and structural similarity index (SSIM) measurements with two different reference data for the resulting clusters [7]. In image processing, PSNR and SSIM ordinarily measure the correctness of a constructed image (the higher the PSNR is, the higher the correctness). PSNR is also used for the measurement of the correctness of the edge detection and the clustering processes. In this study, each pixel in a cluster takes the mean value of the NDVs of the pixels in the cluster after the clustering process. The PSNR and SSIM scores of clustering results are evaluated by referring to the source NDVs and the ground-truth (GT) semantic clusters (each pixel in the GT has the mean value of the NDVs of the cluster which it belongs to), separately. Finally, we give the results of the accuracy measurements under the guidance of the labeled forms of the GT images. Looking at the accuracy results, the NDV-based success measurement methods can be evaluated in terms of reliability, besides measuring the success of the clustering outputs in an additional way.

2. Background

2.1. Dataset

In this study, 10 high-resolution remote sensing images are tested such that the size of each is 6000×6000 pixels. The images are taken from the Potsdam dataset included in ISPRS WG II/4 (2019) [1]. The 10 images have the attributes that consist of R, G, B, IR, and DSM. The contest also provides the normalized DSM (nDSM) values

---

of the pixels. In Figure 1, a remote sensing image included in the dataset is presented with RGB, RGIR (red, green, and infrared), IRRG (infrared, red, and green), and nDSM attribute values.
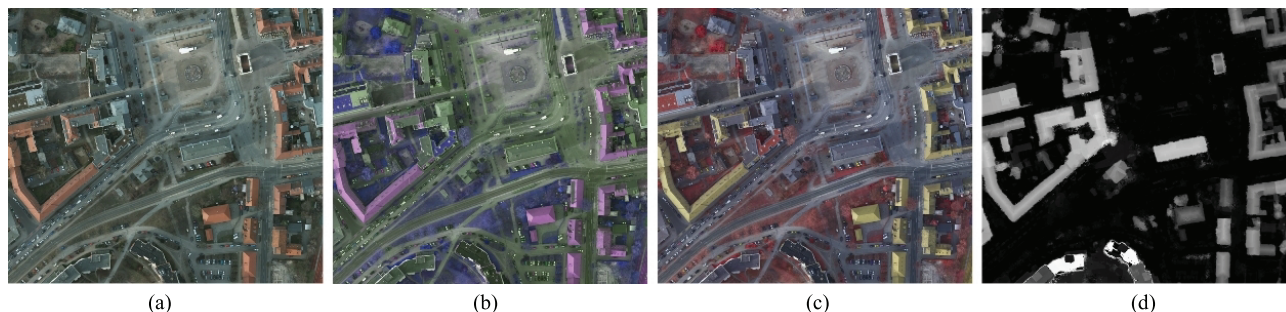

| (a) | (b) | (c) | (d) |

Figure 1. The presentations of the image "Potsdam_6_10" with RGB (a), RGIR (b), IRRG (c), and nDSM (d) images.

2.2. Color transformation

Each attribute of a pixel in a multi-spectral image determines the location of the element through an axis in a multidimensional space. Color transformation transfers a pixel from a space to another space by some mathematical calculations. This transformation is commonly required to enhance the discrimination of the pixels that should belong to different clusters or classes depending on the human interpretation.

The remote sensing images used in this study come in RGB colors and IR value. The human perception appropriate color spaces L*a*b* and the graphical design appropriate color space HSV are also tested besides RGB, IRRG, RGIR color spaces and their gray-level transformation in this paper. The order difference between IRRG and RGIR is important for the color transformations. For example, in the transformation from IRRG to HSV, the values IR, R, and G are considered as if they are R, G, and B respectively.

2.2.1. Gray-level color transformation

Gray-level color is one of the color attributes such that it ranges between black and white colors [e.g., $0 - 255$]. The gray-level values can be obtained with luminance sensing devices directly or the transformation of color values to gray-level values. In many cases, the gray-level values are seen as the luminance values. Some color systems use these luminance values in its space such as YIQ, YUV, and L*a*b*. Luminance values are obtained from the RGB color values as in Eq. (1) [8, 9]:

$$Gray = \begin{bmatrix} 0.299 & 0.587 & 0.114 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{1}$$

2.2.2. L*a*b* color transformation

The axes of the L*a*b* color space indicates the values of lightness (L*), the colors in the range between green and red (a*), and the colors in the range between blue and yellow (b*) [10]. To indicate a pixel RGB color vector in L*a*b* space, firstly RGB values are transferred to XYZ color space; then, the XYZ values are transferred

to L*a*b* color space as in Eqs. (2) and (3) [9]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \tag{2}$$

$$L* = 116.\left(\sqrt[3]{\frac{Y}{Y_0}}\right) - 16, \quad a* = 500.\left(\sqrt[3]{\frac{X}{X_0}} - \sqrt[3]{\frac{Y}{Y_0}}\right), \quad b* = 200.\left(\sqrt[3]{\frac{Y}{Y_0}} - \sqrt[3]{\frac{Z}{Z_0}}\right). \tag{3}$$

The values X_0, Y_0, and Z_0 indicate the values X, Y, and Z of the standard white, respectively. In this study, X_0, Y_0, and Z_0 are used as the values 0.9504, 1.0000, and 1.0888, respectively; such that, those simulate noon daylight with the correlated color temperature of 6504 K. These characteristics indicate the white color in the D65 reference that is the default for input RGB color transformation in the MATLAB program.

2.2.3. HSV color transformation

The axes of the HSV color space indicates the values of hue (H), saturate (S), and value (V) [11, 12]. In this space, the chromatic values can be changed with only the value H. The transformation of RGB values to HSV color space is performed as in Eqs. (4)–(6) [13]:

$$H = \cos^{-1}\frac{\frac{1}{2}.\left[(R-G)+(R-B)\right]}{\sqrt{(R-G)^2 + (R-G).(G-B)}}, \tag{4}$$

$$S = 1 - \frac{3}{R+G+B}.min\,(R,G,B), \tag{5}$$

$$V = \frac{1}{3}.(R+G+B). \tag{6}$$

In this study, RGIR and IRRG values are also tested by converting them to L*a*b* (RGIR→L*a*b* and IRRG→L*a*b*) and HSV (RGIR→HSV and IRRG→HSV) spaces as if they are RGB values respecting to the color orders.

2.3. Distance norms

A distance norm measures the difference between two points in a coordinate space. Each point has vector values in a multidimensional space. A vector $\vec{x}$ has a dimension with a number of values such that $\vec{x} = \{x_1, x_2, \ldots, x_D\}$ where D is the number of dimension. The numerical and positive value $diff(\vec{x}, \vec{y})$ refers to the distance between the vectors $\vec{x}$ and $\vec{y}$ and $diff(\vec{x}, \vec{y}) = diff(\vec{y}, \vec{x})$ [14]. Distance norms determine how to calculate the distance between $\vec{x}$ and $\vec{y}$ using the values of the vectors.

2.3.1. Manhattan distance (L_1 norm)

According to Manhattan distance (also known as city block distance), the difference between two points is calculated by summing the absolute differences between the correspondence vector values as seen in Eq. (7) [15, 16]:

$$diff(\vec{x}, \vec{y}) = \sum_{i=1}^{D} |x_i - y_i|. \tag{7}$$

2.3.2. Euclidean distance (L_2 norm)

Euclidean distance is the shortest path between two points in the Euclidean space such that it is calculated by taking the square root of the sum of the squared absolute differences between the correspondence vector values as seen in Eq. (8) [14, 17]:

$$diff(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{D} (x_i - y_i)^2}.$$ (8)

2.3.3. Squared Euclidean distance (L_2^2 norm)

Squared Euclidean distance is the case of the Euclidean distance without taking the square as seen in Eq. (9) [14, 17]:

$$diff(\vec{x}, \vec{y}) = \sum_{i=1}^{D} (x_i - y_i)^2.$$ (9)

3. Methods

3.1. Normalized distance values (NDVs)

The NDVs of a point in a multidimensional space is the normalized and rounded distance from the point to the center of all points in a specified space [4]. NDVs are used to reduce the vector values that consist of more than one coordinate values into one value. This method has been proposed to process the multidimensional data on a histogram because applying some methods (e.g., the k-means clustering) on very large amounts of data with less processing needs a histogram-based approach. A histogram indicates the frequencies of the data elements having the same attribute values [3, 4]. The attribute values in a histogram must be normalized in a specified range (the closed range $0 - 255$ is used in this study). If the values are discretized into specified bins, the processes on a histogram are faster (the values are discretized into the nearest integers in this study). The calculation of the NDV of the vector $\vec{x}$ is denoted in Eqs. (10) and (11) for the normalization range is the closed range 0–255 and the sampling ranges for discretization are integers.

$$diff = diff(\vec{x}, \vec{\mu}),$$ (10)

$$NDV_{\vec{x}} = round\left(255.\frac{diff - diff_{min}}{diff_{max} - diff_{min}}\right).$$ (11)

In Eq. (10), $\vec{\mu}$ refers to the vector of the mean feature values of all points in the data. In Eq. (11), $diff_{min}$ and $diff_{max}$ values are the distance values of the nearest and the farthest elements to the center, respectively. The function $round$ turns a distance value to the nearest integer. In this study, the NDV transformation is applied using different attribute values and different distance norms. In Figure 2, some of the NDVs with different attributes and distance norms seem visually.

3.2. Histogram-based fast k-means clustering

K-means algorithm is one of the well-known clustering algorithms. This algorithm initially needs the number of clusters and runs iteratively. At the start, the algorithm allocates the cluster centers randomly or using specific
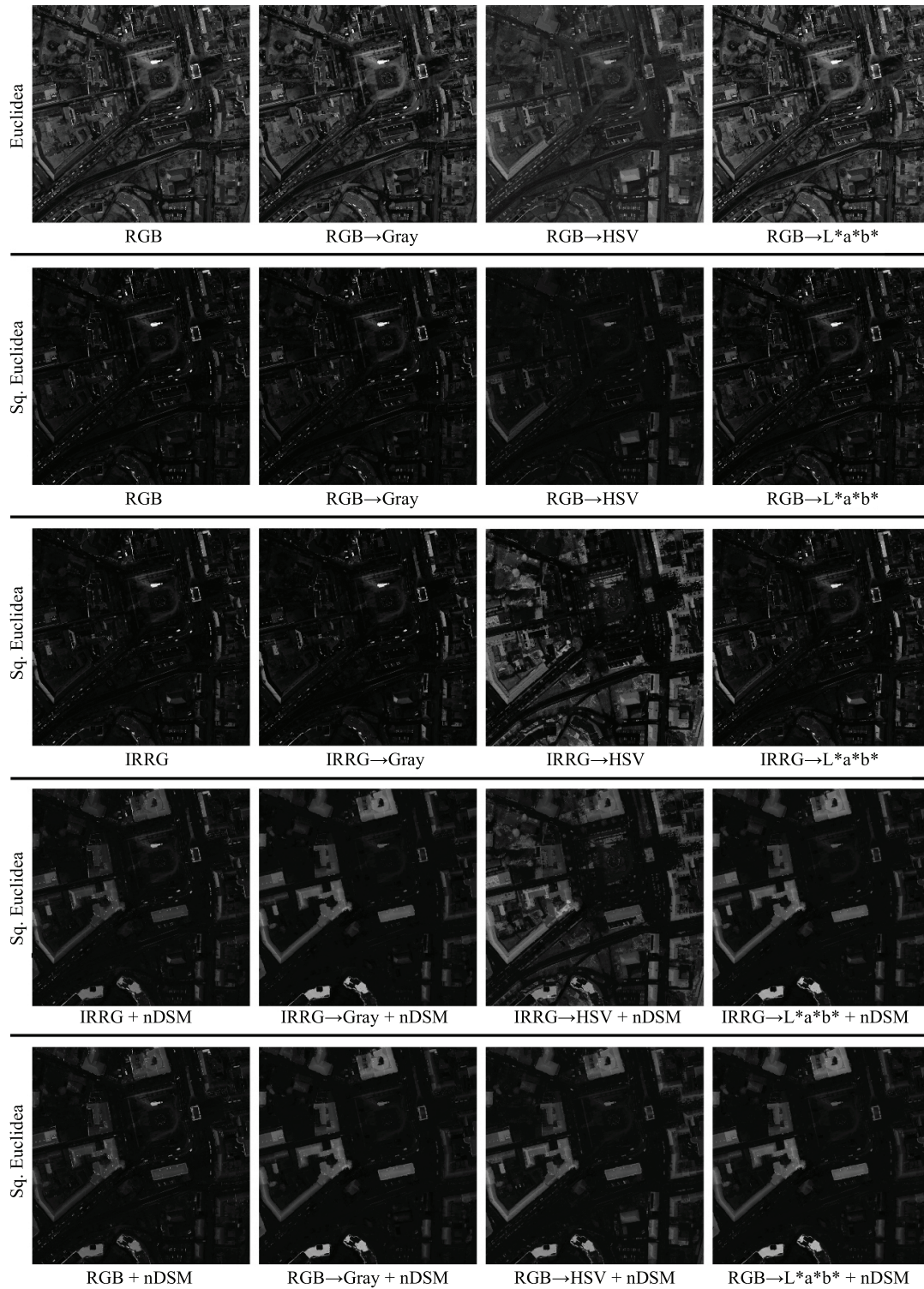
Figure 2. The presentations of the normalized distance values with some attributes and distance metrics.

techniques. Each data element is appointed to the nearest cluster center. After the appointment process, each cluster center is reallocated to the mean position value/s of the elements in the cluster. In the next iteration,

all the elements are appointed to the reallocated cluster centers; and then, the cluster centers are reallocated to the new mean position values in the clusters. The clustering process ends when there is no change at the cluster centers or the iteration number reaches the specified upper iteration limit.

The k-means algorithm runs very slowly on a huge amount of data because the distance between all the elements and the cluster centers are recalculated in each iteration. According to the histogram-based approach, when an element is appointed to the nearest cluster center, all the elements in the same histogram bin with the element are appointed to the same cluster center. However, this approach needs the discretized and normalized one-dimensional attribute values. The NDV method ensures these requirements for the multidimensional attribute values. The general framework of the clustering algorithm is sketched as follows:

Step 1- Specify the k cluster number.

Step 2- Allocate the k cluster centers (e.g., randomly).

Step 3- Assign the histogram attribute values to the nearest cluster centers.

Step 4- Appoint all the elements to the clusters according to the assigned attribute values.

Step 5- Reallocate the cluster centers to the mean position values of the elements in the cluster.

Step 6- If there is no change at the cluster centers or the iteration number reaches the specified upper iteration limit, the algorithm ends. Otherwise, go to Step 3.

4. Comparison metrics

4.1. Peak signal-to-noise rate (PSNR)

PSNR is the ratio of the maximum possible signal power to the power of deterioration noise [7, 18]. It can be also defined as the logarithmic function of the rate of peak signal value to mean square error (MSE) [19]. The formulation of the PSNR for the $M \times N$ sized image signal is seen in Eqs. (12) and (13):

$$MSE = \frac{\sum\limits_{i=1}^{M} \sum\limits_{j=1}^{N} [I_1(i,j) - I_2(i,j)]^2}{M.N}, \tag{12}$$

$$PSNR = 10 \log_{10}\left(\frac{R^2}{MSE}\right). \tag{13}$$

In Eq. (12), I_1 is the source image, I_2 is the result image, and R is the maximum possible value in Eq.(13) (e.g., 255).

PSNR is generally used to measure the reconstruction success in image compression and noise reduction. PSNR is also used to measure the correctnesses of edge detection and clustering methods in some cases [18, 20, 21].

4.2. Structural similarity index (SSIM)

SSIM is another method used to evaluate the quality of result image [22]. SSIM extracts the structural information from the source and result images, and compares them to each other [23]. The formulation of the SSIM for the corresponding pixel pairs of the images I_1 and I_2 is seen in Eq. (14):

$$SSIM(I_1(i,j), I_2(i,j)) = \frac{(2.\mu_1.\mu_2 + C_1).(2.\sigma_{1,2} + C_2)}{(\mu_1^2 + \mu_2^2 + C_1).(\sigma_1^2 + \sigma_2^2 + C_2)}. \tag{14}$$

In Eq. (14), μ_1 and μ_2 are the local mean feature values, while σ_1 and σ_2 are the local standard deviations of the corresponding local features, respectively. Local features are included in a local region that surrounds the related pixel with a radius value such as 1.5 (default value in MATLAB). $\sigma_{1,2}$ refers to the cross-covariance between the corresponding local features. C_1 and C_2 are very small positive constant values that prevent the zero values in the equation. The overall SSIM value of the images I_1 and I_2 can be calculated by obtaining the mean value of the SSIM values of all of the pixel pairs as seen in Eq. (15):

$$SSIM = \frac{\sum\limits_{i=1}^{M} \sum\limits_{j=1}^{N} SSIM\left(I_1\left(i,j\right), I_2\left(i,j\right)\right)}{M.N}. \tag{15}$$

4.3. Accuracy

Accuracy is one of the most commonly used measurement methods to assess the success of the classification results. Accuracy is the ratio of correctly classified elements to all elements. The labels of an element both in the reference and the output data are compared to determine if an element is correctly classified [24]. However, the clustering outputs can assign the output labels independently of the reference data, because the k-means clustering process is executed unsupervised. In such a case, the majority label pairing is one of the methods used in literature [25, 26]. In our implementation, the labels in the reference data (labeled GT in our experiments) are paired with the labels in the output data (labeled result image in our experiments) to provide maximum overlap. After the pairing, the accuracy values of the clustering results can be computed as in the classification assessment.

5. Clustering results

We applied the clustering process to 10 high-resolution images (6000×6000). These images are selected among those with GT images from the Potsdam dataset.

NDVs are taken from different attribute values (also transformed into different color spaces) with different distance norms. The histogram-based k-means algorithm has been applied to NDVs with the upper limit iteration number 50 [4]). The number of clusters has been selected by considering the number of classes in GT images. After the clustering process, the 5×5 median filter has been applied to the result labels as postprocessing to remove small cluster particles in the experiments. In Figure 3, an example is seen for the first clustered result and its median filtered result.
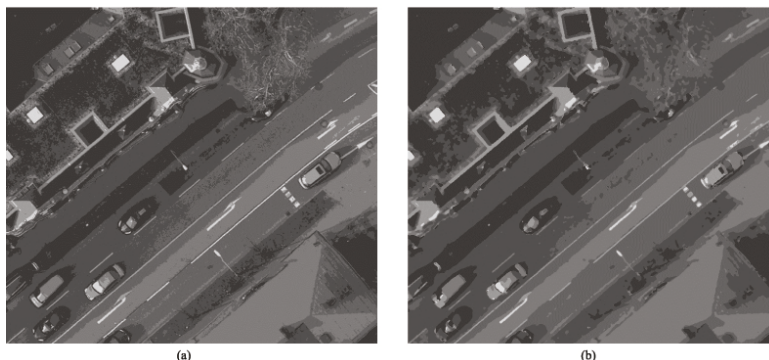

Figure 3. First clustered image (a) colored with the mean NDVs and its median filtered result (b).

In the PSNR calculation, the mean NDVs of the pixels in the result clusters is used as result image and two images are used as source image separately. One is the matrix that consists of the NDVs and the other is the matrix that consists of the mean NDVs of the pixels of the true classes under the guidance of the GT images. The visual example is shown in Figure 4.
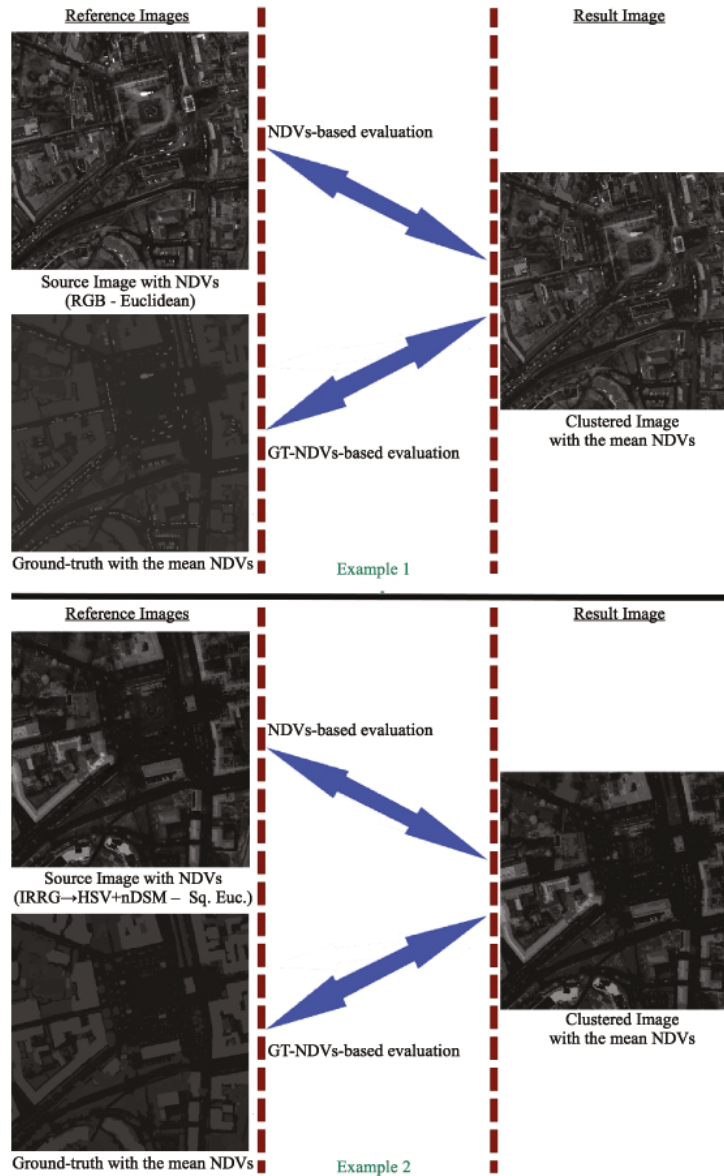

Figure 4. Two examples of reference and result images for two different evaluations.

The PSNR comparisons (NDVs-based and GT-NDVs-based) of the mean clustering results of the 10 images and the standard deviations are seen in Table 1. Some of the clustering results are visually presented as randomly colored in Figure 5. In Figure 5, the clustering results of the image "Potsdam_6_10" are presented, because the PSNR scores of this image are the nearest to the mean PSNR scores of the 10 images. The SSIM

comparisons (NDVs-based and GT-NDVs-based) and the accuracy results of the mean clustering results of the 10 images and the standard deviations are seen in Table 2 and 3, respectively.
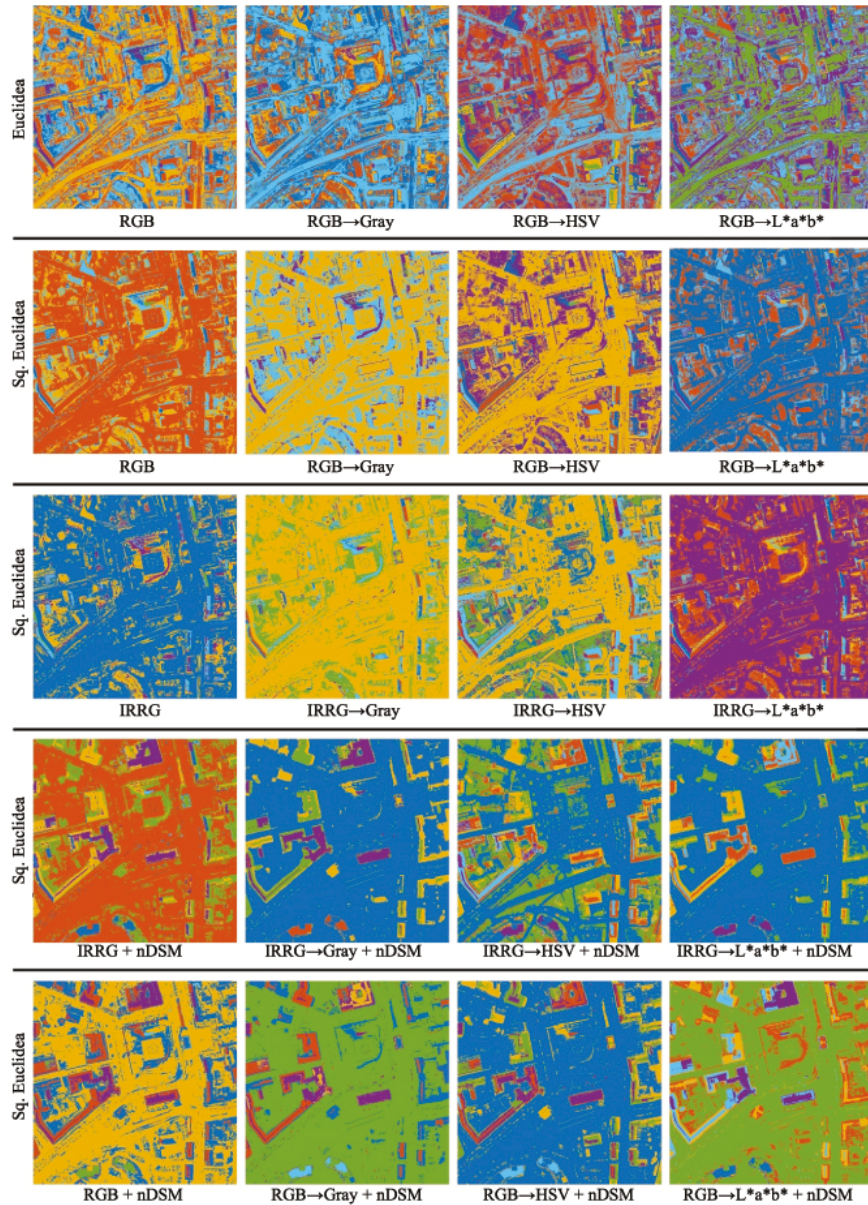

Figure 5. The clustering results of the image "Potsdam_6_10" with some attributes, color transformations, and distance norms.

6. Conclusion

In this study, a histogram-based fast k-means algorithm is applied to 10 high-resolution (6000×6000) images using normalized distance values (NDVs) with different attribute values (color and nDSM values) and distance norms. The images are selected from the Potsdam dataset with semantic clustering ground-truths. The images in the dataset come with the attributes red (R), green (G), blue (B), the infrared (IR) and the normalized digital surface maps (nDSM) values. In this work, the different combinations of the attributes within the Gray,

Table 1. The PSNR comparisons (NDVs-based and GT-NDVs-based) of the mean clustering results of the 10 images and the standard deviations.

	PSNR based on NDVs			PSNR based on mean GT-NDVs		
	Manhattan	Euclidean	Sq. Euclidean	Manhattan	Euclidean	Sq. Euclidean
RGB	25.4 ± 0.4	25.7 ± 0.9	29.5 ± 1.0	18.8 ± 0.7	18.9 ± 0.7	22.7 ± 1.1
RGB→Gray	24.9 ± 0.8	25.1 ± 0.8	29.4 ± 0.9	18.6 ± 0.7	18.5 ± 0.7	22.7 ± 1.2
RGB→HSV	25.8 ± 0.8	28.1 ± 0.7	32.6 ± 1.0	22.2 ± 1.4	21.9 ± 1.2	26.1 ± 1.8
RGB→L*a*b*	25.0 ± 0.8	25.0 ± 0.7	29.0 ± 1.2	18.5 ± 0.9	18.2 ± 0.6	21.8 ± 1.0
RGIR	25.8 ± 0.8	26.1 ± 0.8	29.8 ± 1.4	19.3 ± 0.7	19.3 ± 0.8	23.2 ± 1.0
RGIR→Gray	25.3 ± 0.8	25.0 ± 0.9	29.6 ± 1.1	18.6 ± 0.7	18.7 ± 0.7	23.0 ± 1.0
RGIR→HSV	27.6 ± 0.7	28.0 ± 0.9	29.8 ± 1.0	20.3 ± 0.6	20.3 ± 0.7	22.5 ± 1.1
RGIR→L*a*b*	25.6 ± 0.8	25.7 ± 0.9	28.6 ± 0.9	18.8 ± 0.5	18.7 ± 0.6	22.0 ± 1.0
IRRG	25.8 ± 0.9	26.0 ± 0.8	29.9 ± 1.3	19.3 ± 0.8	19.3 ± 0.8	23.1 ± 1.0
IRRG→Gray	25.7 ± 1.1	25.5 ± 0.8	30.0 ± 1.4	18.6 ± 0.7	18.6 ± 0.8	22.9 ± 1.1
IRRG→HSV	25.2 ± 1.1	23.5 ± 1.6	23.9 ± 2.4	18.0 ± 0.6	16.5 ± 0.3	18.2 ± 0.9
IRRG→L*a*b*	27.8 ± 0.8	25.7 ± 0.8	29.3 ± 1.3	20.3 ± 0.9	18.6 ± 0.8	22.1 ± 1.3
RGB+nDSM	27.5 ± 0.9	27.5 ± 0.8	31.1 ± 0.8	20.3 ± 0.6	19.8 ± 0.4	23.1 ± 0.8
RGB→Gray+nDSM	29.3 ± 0.7	28.8 ± 0.4	32.7 ± 1.2	20.5 ± 0.9	19.4 ± 1.1	22.1 ± 2.1
RGB→HSV+nDSM	29.1 ± 0.9	29.0 ± 0.5	32.1 ± 1.0	21.2 ± 1.5	20.3 ± 1.2	22.4 ± 1.9
RGB→L*a*b*+nDSM	29.0 ± 0.7	28.9 ± 0.5	32.5 ± 1.1	21.1 ± 1.0	19.3 ± 1.1	21.9 ± 2.1
RGIR+nDSM	28.0 ± 0.7	28.1 ± 0.6	31.4 ± 1.0	20.6 ± 0.7	20.0 ± 0.5	23.1 ± 1.1
RGIR→Gray+nDSM	29.1 ± 0.6	28.7 ± 0.5	32.6 ± 1.3	20.5 ± 1.0	19.4 ± 1.1	22.1 ± 2.1
RGIR→HSV+nDSM	29.2 ± 0.7	29.2 ± 0.4	31.6 ± 0.9	20.8 ± 0.9	20.1 ± 1.1	21.8 ± 1.9
RGIR→L*a*b*+nDSM	28.8 ± 0.6	28.9 ± 0.5	32.5 ± 1.2	20.1 ± 1.1	19.5 ± 1.1	21.8 ± 2.2
IRRG+nDSM	27.8 ± 0.7	28.0 ± 0.7	31.4 ± 0.9	20.6 ± 0.7	20.6 ± 0.6	23.2 ± 1.1
IRRG→Gray+nDSM	29.3 ± 0.7	29.2 ± 0.4	32.7 ± 1.3	20.3 ± 1.1	19.3 ± 1.2	21.9 ± 2.2
IRRG→HSV+nDSM	27.2 ± 0.8	25.5 ± 1.2	27.0 ± 1.5	19.5 ± 1.0	18.2 ± 0.6	19.9 ± 1.0
IRRG→L*a*b*+nDSM	29.2 ± 0.5	29.0 ± 0.6	32.6 ± 1.3	21.1 ± 1.1	19.4 ± 1.2	21.7 ± 2.2
RGB+IR	25.9 ± 0.8	25.9 ± 0.8	29.3 ± 1.0	19.5 ± 0.7	19.5 ± 0.7	23.3 ± 1.0
RGB→Gray+IR	25.9 ± 0.8	26.0 ± 0.9	29.8 ± 1.3	19.4 ± 0.8	19.5 ± 0.8	23.5 ± 1.0
RGB→HSV+IR	28.2 ± 0.7	27.9 ± 0.6	31.7 ± 1.1	21.4 ± 1.4	21.1 ± 1.1	24.7 ± 1.7
RGB→L*a*b*+IR	25.9 ± 0.8	25.7 ± 0.6	29.4 ± 1.3	19.0 ± 0.9	19.3 ± 0.7	23.0 ± 1.0
RGB+IR+nDSM	27.2 ± 1.0	27.5 ± 0.9	30.7 ± 0.9	20.6 ± 0.7	20.1 ± 0.5	23.3 ± 0.9
RGB→Gray+IR+nDSM	28.5 ± 0.7	28.5 ± 0.4	32.0 ± 1.0	20.9 ± 0.7	20.0 ± 0.7	22.9 ± 1.6
RGB→HSV+IR+nDSM	28.8 ± 0.7	28.6 ± 0.5	31.5 ± 1.0	21.0 ± 1.1	20.3 ± 0.8	22.6 ± 1.6
RGB→L*a*b*+IR+nDSM	28.4 ± 0.4	28.4 ± 0.5	31.9 ± 1.0	20.6 ± 0.8	19.9 ± 0.7	22.7 ± 1.6

HSV and L*a*b* color transformations are used. Besides them, the Manhattan, the Euclidean, and the squared Euclidean distance metrics are used for the NDV calculations separately. The PSNR and SSIM measurements are carried out within two different ways. As the source image, one calculation uses the NDVs and the other one uses the semantic labeling ground-truth, although the clustering process is not done semantically. Additionally,

Table 2. The SSIM comparisons (NDVs-based and GT-NDVs-based) of the mean clustering results of the 10 images and the standard deviations.

	SSIM based on NDVs			SSIM based on mean GT-NDVs		
	Manhattan	Euclidean	Sq. Euclidean	Manhattan	Euclidean	Sq. Euclidean
RGB	0.65 ± 0.03	0.66 ± 0.04	0.74 ± 0.03	0.61 ± 0.03	0.61 ± 0.03	0.66 ± 0.03
RGB→Gray	0.61 ± 0.03	0.62 ± 0.03	0.72 ± 0.01	0.59 ± 0.03	0.58 ± 0.03	0.63 ± 0.04
RGB→HSV	0.69 ± 0.02	0.69 ± 0.02	0.82 ± 0.01	0.69 ± 0.02	0.69 ± 0.02	0.78 ± 0.02
RGB→L*a*b*	0.62 ± 0.03	0.62 ± 0.03	0.71 ± 0.02	0.61 ± 0.03	0.59 ± 0.03	0.62 ± 0.03
RGIR	0.69 ± 0.03	0.69 ± 0.03	0.77 ± 0.03	0.65 ± 0.03	0.65 ± 0.02	0.69 ± 0.05
RGIR→Gray	0.62 ± 0.03	0.62 ± 0.03	0.72 ± 0.02	0.58 ± 0.02	0.58 ± 0.03	0.62 ± 0.04
RGIR→HSV	0.69 ± 0.02	0.72 ± 0.02	0.80 ± 0.02	0.67 ± 0.02	0.70 ± 0.02	0.73 ± 0.02
RGIR→L*a*b*	0.65 ± 0.03	0.67 ± 0.03	0.75 ± 0.02	0.63 ± 0.02	0.64 ± 0.03	0.66 ± 0.04
IRRG	0.69 ± 0.02	0.69 ± 0.03	0.77 ± 0.03	0.64 ± 0.04	0.66 ± 0.02	0.69 ± 0.03
IRRG→Gray	0.64 ± 0.02	0.63 ± 0.02	0.73 ± 0.02	0.58 ± 0.03	0.58 ± 0.02	0.63 ± 0.04
IRRG→HSV	0.65 ± 0.02	0.66 ± 0.03	0.66 ± 0.03	0.60 ± 0.02	0.61 ± 0.02	0.58 ± 0.05
IRRG→L*a*b*	0.71 ± 0.02	0.68 ± 0.03	0.75 ± 0.02	0.68 ± 0.03	0.64 ± 0.02	0.65 ± 0.04
RGB+nDSM	0.72 ± 0.04	0.76 ± 0.04	0.83 ± 0.02	0.70 ± 0.03	0.74 ± 0.04	0.74 ± 0.04
RGB→Gray+nDSM	0.78 ± 0.02	0.82 ± 0.02	0.86 ± 0.01	0.77 ± 0.01	0.80 ± 0.03	0.80 ± 0.03
RGB→HSV+nDSM	0.72 ± 0.02	0.76 ± 0.03	0.85 ± 0.01	0.74 ± 0.01	0.77 ± 0.02	0.82 ± 0.01
RGB→L*a*b*+nDSM	0.77 ± 0.01	0.82 ± 0.02	0.86 ± 0.01	0.76 ± 0.02	0.79 ± 0.02	0.79 ± 0.02
RGIR+nDSM	0.75 ± 0.03	0.78 ± 0.04	0.84 ± 0.02	0.73 ± 0.03	0.76 ± 0.03	0.77 ± 0.03
RGIR→Gray+nDSM	0.78 ± 0.02	0.82 ± 0.02	0.86 ± 0.01	0.77 ± 0.02	0.80 ± 0.02	0.80 ± 0.02
RGIR→HSV+nDSM	0.75 ± 0.02	0.79 ± 0.02	0.85 ± 0.01	0.74 ± 0.01	0.78 ± 0.02	0.80 ± 0.01
RGIR→L*a*b*+nDSM	0.77 ± 0.02	0.82 ± 0.02	0.87 ± 0.01	0.76 ± 0.02	0.80 ± 0.02	0.81 ± 0.02
IRRG+nDSM	0.75 ± 0.03	0.78 ± 0.03	0.84 ± 0.02	0.73 ± 0.03	0.75 ± 0.03	0.78 ± 0.03
IRRG→Gray+nDSM	0.79 ± 0.02	0.83 ± 0.02	0.86 ± 0.01	0.77 ± 0.01	0.80 ± 0.02	0.81 ± 0.02
IRRG→HSV+nDSM	0.71 ± 0.03	0.73 ± 0.04	0.76 ± 0.04	0.69 ± 0.02	0.71 ± 0.03	0.69 ± 0.04
IRRG→L*a*b*+nDSM	0.78 ± 0.02	0.83 ± 0.02	0.86 ± 0.01	0.78 ± 0.02	0.80 ± 0.02	0.81 ± 0.02
RGB+IR	0.68 ± 0.03	0.69 ± 0.03	0.76 ± 0.02	0.65 ± 0.03	0.67 ± 0.02	0.70 ± 0.04
RGB→Gray+IR	0.69 ± 0.02	0.68 ± 0.03	0.76 ± 0.02	0.66 ± 0.03	0.65 ± 0.02	0.69 ± 0.04
RGB→HSV+IR	0.70 ± 0.02	0.71 ± 0.02	0.82 ± 0.01	0.70 ± 0.02	0.70 ± 0.02	0.77 ± 0.03
RGB→L*a*b*+IR	0.68 ± 0.03	0.68 ± 0.03	0.75 ± 0.02	0.65 ± 0.03	0.66 ± 0.03	0.68 ± 0.03
RGB+IR+nDSM	0.73 ± 0.04	0.76 ± 0.04	0.83 ± 0.02	0.71 ± 0.03	0.75 ± 0.03	0.75 ± 0.04
RGB→Gray+IR+nDSM	0.77 ± 0.02	0.79 ± 0.03	0.85 ± 0.02	0.75 ± 0.02	0.78 ± 0.03	0.79 ± 0.02
RGB→HSV+IR+nDSM	0.73 ± 0.02	0.76 ± 0.03	0.84 ± 0.02	0.74 ± 0.02	0.76 ± 0.03	0.80 ± 0.01
RGB→L*a*b*+IR+nDSM	0.76 ± 0.02	0.79 ± 0.03	0.85 ± 0.01	0.75 ± 0.02	0.77 ± 0.03	0.78 ± 0.02

accuracy measurement results are acquired with the most overlapping pairing between the labels in the labeled GT and the labeled clustering output. Looking at the outperformed results, it seems that the evaluating methods are frequently consistent. Considering this case, it is concluded that the clustering success of such images used in this study can be assessed by the NDV-based evaluation, without the need of GT images. As the distance metric, the squared Euclidean distance is the most effective one. As the color system, HSV gets the most successful results with both the RGB to HSV and RGIR to HSV transformation. When the nDSM value is added, Gray and L*a*b* transformations also show good success. It also seems that the attribute value IR

Table 3. Accuracy comparison of the mean clustering results of the 10 images and the standard deviations.

| | Accuracy based on labeled GT | | |
	Manhattan	Euclidean	Sq. Euclidean
RGB	0.31 ± 0.04	0.30 ± 0.04	0.36 ± 0.07
RGB→Gray	0.29 ± 0.03	0.29 ± 0.03	0.36 ± 0.07
RGB→HSV	0.31 ± 0.05	0.31 ± 0.05	0.40 ± 0.09
RGB→L*a*b*	0.30 ± 0.03	0.31 ± 0.04	0.35 ± 0.07
RGIR	0.29 ± 0.02	0.30 ± 0.03	0.35 ± 0.06
RGIR→Gray	0.28 ± 0.03	0.29 ± 0.02	0.35 ± 0.07
RGIR→HSV	0.30 ± 0.02	0.33 ± 0.03	0.39 ± 0.08
RGIR→L*a*b*	0.29 ± 0.03	0.33 ± 0.04	0.36 ± 0.06
IRRG	0.28 ± 0.04	0.31 ± 0.03	0.36 ± 0.07
IRRG→Gray	0.28 ± 0.03	0.27 ± 0.02	0.35 ± 0.07
IRRG→HSV	0.35 ± 0.04	0.36 ± 0.02	0.45 ± 0.08
IRRG→L*a*b*	0.30 ± 0.04	0.29 ± 0.04	0.35 ± 0.07
RGB+nDSM	0.31 ± 0.04	0.36 ± 0.05	0.40 ± 0.08
RGB→Gray+nDSM	0.34 ± 0.05	0.37 ± 0.06	0.41 ± 0.09
RGB→HSV+nDSM	0.36 ± 0.05	0.39 ± 0.07	0.42 ± 0.09
RGB→L*a*b*+nDSM	0.34 ± 0.06	0.38 ± 0.08	0.41 ± 0.08
RGIR+nDSM	0.32 ± 0.03	0.36 ± 0.05	0.39 ± 0.08
RGIR→Gray+nDSM	0.34 ± 0.05	0.37 ± 0.08	0.41 ± 0.09
RGIR→HSV+nDSM	0.34 ± 0.03	0.40 ± 0.06	0.37 ± 0.13
RGIR→L*a*b*+nDSM	0.36 ± 0.05	0.39 ± 0.06	0.41 ± 0.08
IRRG+nDSM	0.32 ± 0.04	0.36 ± 0.05	0.40 ± 0.08
IRRG→Gray+nDSM	0.35 ± 0.05	0.39 ± 0.05	0.41 ± 0.08
IRRG→HSV+nDSM	0.36 ± 0.05	0.42 ± 0.05	0.48 ± 0.06
IRRG→L*a*b*+nDSM	0.35 ± 0.07	0.37 ± 0.05	0.41 ± 0.09
RGB+IR	0.31 ± 0.04	0.32 ± 0.03	0.37 ± 0.07
RGB→Gray+IR	0.30 ± 0.03	0.30 ± 0.03	0.35 ± 0.07
RGB→HSV+IR	0.33 ± 0.04	0.31 ± 0.04	0.39 ± 0.08
RGB→L*a*b*+IR	0.31 ± 0.04	0.29 ± 0.04	0.35 ± 0.06
RGB+IR+nDSM	0.33 ± 0.04	0.37 ± 0.05	0.40 ± 0.08
RGB→Gray+IR+nDSM	0.34 ± 0.03	0.37 ± 0.05	0.40 ± 0.08
RGB→HSV+IR+nDSM	0.35 ± 0.04	0.36 ± 0.06	0.38 ± 0.12
RGB→L*a*b*+IR+nDSM	0.34 ± 0.04	0.36 ± 0.04	0.40 ± 0.08

hardly affects the performance of the clustering process for urban images when it is added to the color values Gray, RGB, HSV, and L*a*b*.

References

[1] Jain AK. Data clustering: 50 years beyond k-means. Pattern Recognition Letters 2010; 31 (8): 651-666. doi: 10.1016/j.patrec.2009.09.011

[2] de Amorim RC, Makarenkov V. Applying subclustering and Lp distance in weighted k-means with distributed centroids. Neurocomputing 2016; 173: 700-707. doi: 10.1016/j.neucom.2015.08.018

[3] Lin CH, Chen CC, Lee HL, Liao JR. Fast k-means algorithm based on a level histogram for image retrieval. Expert Systems with Applications 2014; 41 (7): 3276-3283. doi: 10.1016/j.eswa.2013.11.017

[4] Baykan NA, Saglam A. Fast k-means color image clustering with normalized distance values. Selcuk University Journal of Engineering, Science and Technology (SUJEST) 2018; 6 (2): 175-187. doi: 10.15317/Scitech.2018.124

[5] Volpi M, Tuia D. Dense semantic labeling of subdecimeter resolution images with convolutional neural networks. IEEE Transactions on Geoscience and Remote Sensing 2017; 55: 881-893.

[6] Marmanis D, Schindler K, Wegner JD, Galliani S, Datcu M et al. Classification with an edge: Improving semantic image segmentation with boundary detection. ISPRS Journal of Photogrammetry and Remote Sensing 2018; 135: 158-172. doi: 10.1109/TGRS.2016.2616585

[7] Sarika KS, Sudha P. An analysis of edge extraction for MRI medical images through mathematical morphological operators approaches. International Journal of Computer Applications 2013; ICRTCT (1): 9-13.

[8] Benedetti L, Corsini M, Cignoni P, Callieri M, Scopigno R. Color to gray conversions in the context of stereo matching algorithms: An analysis and comparison of current methods and an ad-hoc theoretically-motivated technique for image matching. Machine Vision and Applications 2012; 23 (2): 327-348. doi: 10.1007/s00138-010-0304-x

[9] Cheng HD, Jiang XH, Sun Y, Wang J. Color image segmentation: Advances and prospects. Pattern Recognition 2001; 34 (12): 2259-2281. doi: 10.1016/S0031-3203(00)00149-7

[10] Margulis D. Photoshop LAB Color: The Canyon Conundrum and Other Adventures in the Most Powerful Colorspace. Berkeley, CA, USA: Peachpit Press, 2006.

[11] Koschan A, Abidi M. Digital Color Image Processing. USA: Wiley-Interscience, 2008.

[12] Baykan NA, Yılmaz N, Kansun G. Case study in effects of color spaces for mineral identification. Scientific Research and Essays 2010; 5 (11): 1243-1253.

[13] Kaur S, Banga VK. Content based image retrieval: Survey and comparison between RGB and HSV model. International Journal of Engineering Trends and Technology (IJETT) 2013; 4 (4): 575-579.

[14] Deza MM, Deza E. Encyclopedia of Distances. Berlin, Germany: Springer-Verlag, 2009.

[15] De Souza RMCR, De Carvalho F de AT. Clustering of interval data based on city-block distances. Pattern Recognition Letters 2004; 25 (3): 353-365. doi: 10.1016/j.patrec.2003.10.016

[16] Gu Z, Zhang Z, Sun J, Li B. Robust image recognition by L1-norm twin-projection support vector machine. Neurocomputing 2017; 223: 1-11. doi: 10.1016/j.neucom.2016.10.008

[17] Perlibakas V, Distance measures for PCA-based face recognition. Pattern Recognition Letters 2004; 25 (6): 711-724. doi: 10.1016/j.patrec.2004.01.011

[18] Poobathy D, Chezian RM. Edge detection operators: Peak signal to noise ratio based comparison. International Journal of Image, Graphics and Signal Processing 2014; 6 (10): 55-61. doi: 10.5815/ijigsp.2014.10.07

[19] Swarnalakshmi R. A Survey on edge detection techniques using different types of digital images. International Journal of Computer Science and Mobile Computing 2014; 3 (7): 694-699.

[20] Karakoyun M, Saglam A, Baykan NA, Altun AA. Non-locally color image segmentation for remote sensing images in different color spaces by using data-clustering methods. In: ICAT'17 5th International Conference on Advanced Technology and Sciences; Istanbul, Turkey; 2017. pp. 6-12.

[21] Harun NH, Izzah M, Ibrahim N, Aziz NS. Comparative study of edge detection algorithm: Vessel wall elasticity measurement for deep vein thrombosis diagnosis. ARPN Journal of Engineering and Applied Sciences 2015; 10 (19): 8635-8641.

[22] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing 2004; 13 (4): 600-612. doi: 10.1109/TIP.2003.819861

[23] Zhang L, Zhang L, Mou X, Zhang D. FSIM: A feature similarity index for image quality assessment. IEEE Transactions on Image Processing 2011; 20 (8): 2378-2386. doi: 10.1109/TIP.2011.2109730

[24] Sokolova M, Lapalme G. A systematic analysis of performance measures for classification tasks. Information Processing and Management 2009; 45 (4): 427-437. doi: 10.1016/j.ipm.2009.03.002

[25] Polak M, Zhang H, Pi M. An evaluation metric for image segmentation of multiple objects. Image and Vision Computing 2009; 27 (8): 1223-1227. doi: 10.1016/j.imavis.2008.09.008

[26] Vo AV, Truong-Hong L, Laefer DF, Bertolotto M. Octree-based region growing for point cloud segmentation. ISPRS Journal of Photogrammetry and Remote Sensing 2015; 104: 88-100. doi: 10.1016/j.isprsjprs.2015.01.011