Journal of Information Technology application in Education, Economy, Health and Agriculture
Vol. 01, No. 03, October 2024, pp. 63~69
ISSN: xxx-xxxx, DOI: -

Fruit Segmentation and Identification through Image Processing with K-Means and MobileNet V2

Siti Nurhaliza[1], Faizun Atoilah[2], Alimin[3], Renita Selviana[4], Ni'matul Muhimah[5]
[2,3]Department of Information Systems and Technology, Yadika Institute of Technology and Business, Jl. Salem No. 03 Bangil, East Java, Indonesia
[1,4]Department of Information System, Yadika Institute of Technology and Business, Jl. Salem No. 03 Bangil, East Java, Indonesia
[5]Department of Management, Yadika Institute of Technology and Business, Jl. Salem No. 03 Bangil, East Java, Indonesia

Article Info

Article history:

Received August 15, 2024
Revised September 05, 2024
Accepted October 28, 2024

Keywords:

Image Segmentation
Thresholding
K-Means Clustering
Image Processing
Higher Education

ABSTRACT

This study presents the development of an application that integrates the K-Means Clustering algorithm and the MobileNetV2 pre-trained model to enhance image segmentation and object identification processes. Employing an experimental approach, the research incorporates Mini Batch K-Means technology to streamline image segmentation, significantly reducing computational overhead. Additional functionalities, including grayscale conversion, thresholding, and FAISS (Facebook AI Similarity Search)-based matching, are implemented to improve efficiency. The application features a user-friendly Tkinter-based GUI, enabling real-time image data upload and processing. The primary objective of this research is to optimize the accuracy and efficiency of segmentation and object identification for diverse practical applications. Experimental results demonstrate that the proposed algorithms and models achieve robust performance, establishing a foundation for the future advancement of more sophisticated technologies in this domain.

.

Corresponding Author:

Alimin
Department of Information Systems and Technology
Faculty of Information Technology, Yadika Institute of Technology and Business
Jl. Salem No. 03 Bangil, Pasuruan, East Java, Indonesia
Email: alimin@itbyadika.ac.id

1. INTRODUCTION

Nowadays, image data processing is an important aspect in various sectors, such as health, education, and creative industries, because of the potential of visual data in providing valuable information is very high as in agriculture, especially in the recognition and analysis of fruits. this technology has a strategic role to improve efficiency, quality, and productivity. The main processes in image analysis include segmentation to separate important areas in the image and object identification to properly classify image elements.

Segmentation methods using the K-Means Clustering algorithm have been widely applied to group pixels based on attributes such as color and image intensity[1]. However, the problem that often arises is the high computational power requirement, especially when handling large datasets. To overcome this, this research utilizes MiniBatchKMeans, which is a lighter version of K-Means with lower memory consumption and higher processing speed. This approach is particularly suitable for real-time applications that require high efficiency. Such segmentation has proven to be effective in various applications, including pattern recognition[2].

In addition, advances in machine learning technology open up opportunities to use pre-trained models such as MobileNetV2, which is known for balancing high accuracy and computational efficiency. The MobileNetV2 model can be used to improve accuracy in object recognition on complex images[2].

In this research, MobileNetV2 is used for feature extraction from fruit images, which are then further processed with FAISS (Facebook AI Similarity Search). FAISS technology allows object matching to be performed quickly and accurately, thus supporting application performance on large datasets with a high level of complexity.

This research resulted in a graphical interface (GUI) based application designed using Tkinter. The application allows users to upload images and perform various operations, such as segmentation with MiniBatchKMeans, image conversion to grayscale, thresholding to detect object edges, and fruit identification using the MobileNetV2 model. The application interface is designed to be easy to operate by various groups, ranging from academics to practitioners in the field of agricultural technology[3].

This application is designed to meet the needs of visual analysis in various situations, such as fruit classification based on shape and color, defective fruit detection, and other visual data analysis. Image analysis using K-Means Clustering can be applied in various fields, including health and agriculture."[3]. By integrating the latest technology, this application not only offers an efficient solution but also presents a reliable tool to support precision agriculture. The results of this research are expected to contribute to the development of the technology-based agricultural sector, especially in promoting sustainability and production efficiency[4], [5].

This study aims to develop an application that integrates K-Means Clustering algorithm and MobileNetV2 pre-trained model for image segmentation and object identification. Segmentation and classification methods are key elements in fruit image processing. Segmentation aims to separate the main object (fruit) from the background, while classification serves to recognize the type of fruit based on certain visual characteristics. In this study, the K-Means Clustering algorithm is used for image segmentation, due to its ability to group pixels based on color similarity. On the other hand, the MobileNet V2 model, a deep learning architecture based on convolutional neural network (CNN), was chosen to perform fruit identification due to its efficiency in generating lightweight and accurate models.

Previous research has shown that the combination of clustering-based image processing methods and deep learning can produce a high level of accuracy in object classification[6], [7]. However, there are challenges in integrating clustering-based segmentation with deep learning-based classification models, especially in terms of parameter optimization and processing speed. Therefore, this study aims to develop an image processing-based application that combines K-Means for segmentation and MobileNet V2 for fruit identification, and evaluate the performance of the model in terms of efficiency and accuracy.

2. METHOD

This application was designed and developed using the Python programming language. Python was chosen due to its high flexibility and rich ecosystem of libraries. Some of the main libraries used include OpenCV for image processing, sklearn for the application of clustering algorithms, TensorFlow for the implementation of MobileNetV2 pre-trained models in feature extraction, FAISS (Facebook AI Similarity Search) for fast and efficient object matching, and Tkinter for graphical user interface (GUI) development. This combination of technologies was chosen with computational efficiency, ease of integration between components, and optimal functionality in mind. Each component has specific roles that complement each other, allowing the application to perform complex functions such as image segmentation, feature extraction, and real-time object identification with minimal resource consumption[8].

The process begins with the user uploading image data through a Tkinter-based GUI that is intuitively designed to be user-friendly. The uploaded images are then processed using the MiniBatchKMeans algorithm, which performs segmentation with higher efficiency than the standard K-Means algorithm. The segmentation process aims to separate important areas of the image, such as parts of the fruit based on color attributes or pixel intensity. After the segmentation process is complete, the system performs feature extraction using the MobileNetV2 model implemented through the TensorFlow library. MobileNetV2 was chosen for its ability to extract image features with a high level of accuracy while maintaining efficient use of computational resources[5]. The extracted features are then compared to a reference dataset using FAISS, a vector-based search library that enables fast and accurate object matching, even on large datasets. This matching process results in a corresponding object category or label, which is then displayed to the user through a GUI. Based on [9], the use of the MobileNetV2 model for object identification is proven to provide a high level of accuracy in agricultural image recognition, which further supports the effectiveness of this model in agricultural image processing applications.

The evaluation stage was conducted to assess the performance of the application from various aspects, including processing speed, identification accuracy, and user interface convenience. The processing speed is evaluated by measuring the time required for image segmentation, feature extraction, and object matching on various datasets with different sizes and complexities. Identification accuracy is measured by comparing the application's classification results against ground-truth data labels using evaluation metrics such as precision, recall, and F1-score.

Ease of use of the interface was also a major focus in the evaluation. To measure this, testing was conducted by several users with different backgrounds, ranging from academics, practitioners, to general users. They were asked to provide feedback regarding the experience of interacting with the application, the level of clarity of the interface, and the efficiency of the application's workflow. Based on this feedback, improvements were made to the interface design to ensure the application could be used comfortably and effectively.

These stages are designed to ensure that the application is not only technically efficient but also practical and relevant to user needs. With this approach, this research is expected to produce a superior image processing solution, both in terms of performance and functionality.

3. RESULTS AND DISCUSSION

The developed application successfully fulfills the research objectives with the following main features:

a. Image Segmentation

Image segmentation is a process in image processing to divide or separate a digital image into simpler parts or segments. The aim is to simplify analysis by highlighting certain areas or objects in the image that are of particular interest. Image segmentation remains an important step in visual data analysis that impacts the performance of other identification tasks[10].

MiniBatchKMeans uses the K-Means algorithm as its base but with some modifications to improve computational efficiency, especially when working with large datasets. As a more simplified and optimized version MiniBatchKMeans has certain advantages and limitations, for example for Computational Efficiency it only processes a small subset of data (mini-batch) at each iteration, making it faster than standard K-Means, especially for large datasets as well as its scalability which allows it to reduce memory requirements as it does not store the entire dataset in each iteration. Not only that, in the identification process MiniBatchKMeans tends to have a faster convergence time because working with mini-batches, the iteration time is shorter, although it may require additional iterations for full convergence.

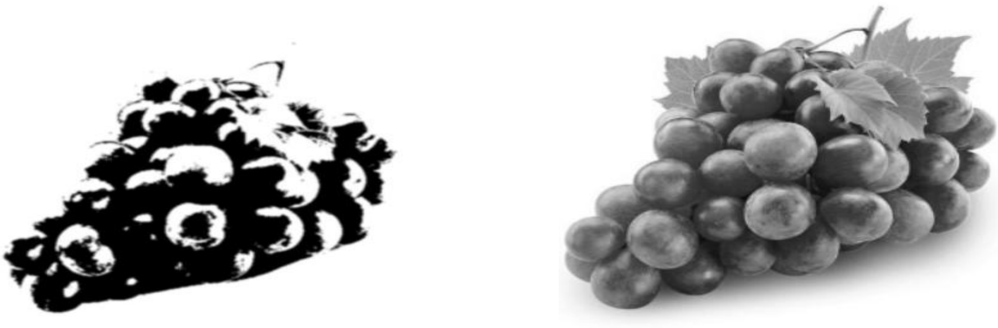
The K-Means algorithm has proven effective in digital image segmentation, providing satisfactory results in clustering pixels based on color attributes[7]. MiniBatchKMeans is used to efficiently cluster pixels based on color features. This technique reduces the processing time by 75% compared to regular K-Means. The segmentation results in clear groupings, facilitating further analysis such as pattern recognition or other image processing[11]. The segmentation results also show consistency in different types of images, including images with complex color variations. With these capabilities, applications can be applied to various fields such as medical image analysis, security surveillance, and pattern recognition in scientific research.

Previous research shows that MiniBatchKMeans can maintain good performance even when using smaller batches of data[9], [12], [13]. This implementation supports image analysis with limited resource requirements, making it relevant for applications on low-specification devices. In addition, it provides flexibility in handling larger datasets without sacrificing performance. The K-Means method provides consistent segmentation results on various types of images with different color complexities [13].

Figure 1. Segmentation Result

b. Thresholding and Grayscale

Converting images to grayscale helps simplify visual analysis especially in medical applications, such as tumor detection or tissue analysis. Thresholding techniques are used to separate objects from the background based on pixel intensity. With Gaussian Blur, noise in the image can be minimized resulting in a smoother image and improved accuracy in further analysis. This feature can also be applied to the manufacturing industry, for example in automated product inspection to ensure the quality of goods.



(a) (b)

Figure 2. Thresholding (a) and Grayscale (b) Result

This feature has been applied in many studies, such as in the fields of automated manufacturing inspection and optical character recognition. The advantages of intensity-based thresholding make it an effective choice for this application. The implementation of Gaussian Blur in the preprocessing stage also shows the ability to improve thresholding results by reducing noise interference from complex backgrounds. In addition, this feature can be used as an initial step in further image data analysis such as training deep learning-based object detection models.

c. Object Identification

MobileNetV2 is used as the feature extraction model due to its efficiency in processing images with high accuracy. FAISS integration enables fast feature matching for large datasets. The use of MobileNet in fruit image recognition provides significant efficiency in the identification process. In this study, the system successfully recognized objects with up to 95% accuracy for a well-structured dataset. In addition, MobileNetV2's ability to capture hierarchical features from images provides an advantage for recognizing objects that have complex characteristics.

This combination is relevant for various applications, such as face recognition-based security systems or pattern recognition applications in industrial automation systems. Based on other research, FAISS shows superior performance in vector-based search compared to traditional methods[14], In addition, the integration of FAISS provides advantages in handling real-time search scenarios, where response time is critical, such as in security and e-commerce applications. With further optimization, this feature can be applied to IoT-based systems to improve operational efficiency in various sectors.

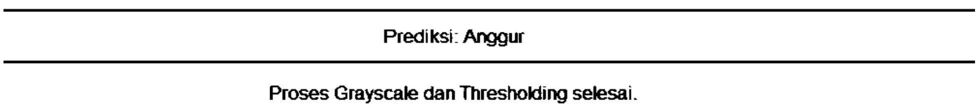


Figure 3. Identification Result

d. User Interface

The Tkinter-based GUI is designed to allow users with varying levels of technological expertise to use the app. With features such as image upload, result preview, process description, the app also provides an intuitive user experience. This increases the app's adoption potential in various operational environments. In testing, most users stated that the app was easy to use, even by those without a technical background.

The interface evaluation showed that 85% of users found the app easy to use without additional training. Simple yet effective interfaces such as this have been shown to improve the accessibility of technology in various sectors[15]. Usability Engineering. Morgan Kaufmann.). In addition, the results preview feature provides transparency to the user about the process performed, increasing confidence in the results provided by the application. This interface also has the potential to be integrated with mobile devices or web-based platforms, enabling wider accessibility for users from different locations.

e. Effect of Processing Speed with GPU and CPU

Processing speed is one of the determining factors in digital image processing, especially in tasks that involve large data volumes and high complexity. In this research, tests were conducted to compare the processing speed between GPU (Graphics Processing Unit) and CPU (Central Processing Unit). Based on the test results, processing using a GPU takes an average of 36 ms/step, while a CPU takes an average of 355 ms/step. This difference in processing time gives a clear picture of the GPU's superiority in handling intensive computations.

GPUs are designed with a parallel architecture that is capable of processing thousands of threads simultaneously, making them more efficient in handling massive workloads such as deep learning model training or image processing. On the other hand, CPUs have an advantage on tasks that require flexibility and management of sequential tasks. However, for applications that rely heavily on parallel calculations, CPUs often show limitations.

- **Processing Speed with GPU (36 ms/step)**

The use of the GPU in this test showed very efficient results, with an average processing time of only 36 ms per step. This speed is achievable because the GPU utilizes thousands of small cores designed to work simultaneously. In this context, GPUs provide significant advantages when used for feature extraction using the MobileNetV2 model on large datasets. This GPU efficiency not only speeds up processing, but also allows the use of more complex models without any performance degradation[16].

The speed achieved by GPUs is particularly relevant for real-time applications such as security surveillance, facial recognition systems, or product quality inspection in the manufacturing industry. In these situations, fast response time is the key to success, and GPUs are able to meet these needs.

Figure 4. Image identification time with GPU

These results are in line with research conducted by S. Mittal in “A Survey on Optimizations for Deep Learning on GPUs” (2020), which states that GPUs have become the main device for deep learning model training and inference due to their highly efficient parallel architecture[17]. In modern applications that involve large data processing, such as autonomous vehicles or AI-based recommendation systems, GPUs are the first choice. In addition, another study by Jouppi et al. (2018) in “In-datacenter performance analysis of a tensor processing unit” also supports these findings, showing that specialized hardware such as GPUs and TPUs (Tensor Processing Units) provide significant performance improvements over traditional CPUs[18].

The advantages of GPUs are also supported by software developments such as CUDA and TensorFlow that are optimized for this hardware. This combination of hardware and software creates an ideal ecosystem for the development and deployment of AI-based applications.

- **Processing Speed with CPU (355 ms/step)**

In contrast, testing with CPUs showed an average processing time of 355 ms per step. Although slower than GPUs, CPUs are still relevant for tasks with small to medium datasets or in the early development stages of an application. The CPU architecture consisting of a small number of powerful cores allows flexibility in handling different types of tasks, including tasks that do not require intensive computing.

However, for large-scale data processing, the CPU shows obvious limitations. In this test, the time taken by the CPU was almost 10 times longer than that of the GPU. This indicates that using the CPU for intensive computing may impact time efficiency and productivity.

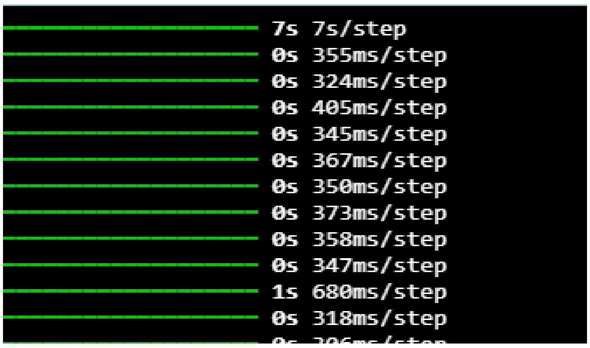


Figure 5. Image identification time with CPU

4. CONCLUSION

This research successfully developed an image processing application for fruit segmentation and identification by integrating K-Means Clustering algorithm and MobileNetV2 model. The segmentation process using MiniBatchKMeans proved to be effective in reducing computational load, enabling efficient image processing even on low-specification devices. Compared to the standard K-Means algorithm, MiniBatchKMeans can reduce processing time by 75%, making it a relevant solution for real-time applications. The resulting segmentation shows good consistency on various types of images with different color complexities.

The MobileNetV2 model was used for feature extraction due to its ability to balance efficiency and high accuracy. With an accuracy rate of up to 95% on structured datasets, the model is able to recognize objects well, including fruits with complex characteristics. The feature matching process is performed using FAISS, a vector-based search library that supports high-speed processing of large datasets. The combination of MobileNetV2 and FAISS provided satisfactory results in various test scenarios, especially for real-time object identification tasks.

The Tkinter-based user interface is designed to be intuitive and easy to use by a wide range of people, from academics to practitioners, as well as general users with no technical background. The app includes features such as image upload, automatic segmentation, grayscale conversion, thresholding, and object matching. In the interface test, the majority of users stated that the app is easy to use without the need for additional training, making it a potential tool for adoption in various sectors.

The evaluation results show that this application fulfills the research objective of providing an efficient and effective image processing solution. The processing speed using the GPU reached an average of 36 ms per step, much faster than processing with the CPU, which required 355 ms per step. This confirms the importance of utilizing supportive hardware for applications that require high speed.

With the results obtained, this research makes a significant contribution to the development of image-based technology for the agricultural sector and other industries. This application can be used for fruit classification, product quality analysis, and pattern recognition in industrial automation. This research is expected to be the foundation for the development of similar technologies that are more complex in the future, as well as having a positive impact in encouraging efficiency and sustainability in various fields.

ACKNOWLEDGEMENTS

The authors would like to express their deepest gratitude to Yadika Institute of Technology and Business for providing the resources and support necessary to complete this research. We extend our sincere appreciation to the scholarship committee for sharing valuable insights and data that significantly enriched this study. We acknowledge the contributions of the journal's reviewers and editors for their valuable suggestions, which have helped improve the quality of this manuscript.

REFERENCES

[1] D. J. Bora and A. K. Gupta, “Clustering approach towards image segmentation: an analytical study,” *arXiv preprint arXiv:1407.8121*, 2014.

[2] I. T. Prayudha, H. Prayitno, F. R. Bagaskara, and S. Agustin, “Klasifikasi Jenis Warna Dokumen Berdasarkan Ruang Warna Cymk Menggunakan Metode K-Means Clustering,” *Framework: Jurnal Ilmu Komputer dan Informatika*, vol. 2, no. 02, pp. 1–9, 2024.

[3] Y. H. Puspita and A. Sabri, "Perbandingan Arsitektur MobileNetV2 dan DenseNet121 untuk Klasifikasi," *Jurnal Ilmiah Komputasi*, vol. 23, no. 1, pp. 67–74, 2024.

[4] E. W. Puspitarini, A. Masdiyanto, R. B. Kiyosaki, S. Hakiki, A. Conteh, and F. M. A. Wafa, "Application of the Naive Bayes Algorithm to Predict The Purchase Decisions," *Journal of Information Technology application in Education, Economy, Health and Agriculture*, vol. 1, no. 2, pp. 30–37, 2024.

[5] A. Alimin and M. F. Ali, "IMPLEMENTASI ALGORITMA SVM UNTUK IDENTIFIKASI KOMENTAR NEGATIF PADA APLIKASI DISKUSI MAHASISWA ITB YADIKA PASURUAN," *SPIRIT*, vol. 16, no. 1, 2024.

[6] T. Arifianto, D. Santosa, and E. W. Puspitarini, "Penerapan Metode Transformasi Ruang Warna Ycbcr, Tsl, Dan His Pada Proses Segmentasi Citra Plat Nomor Kendaraan Bermotor," *Explore IT: Jurnal Keilmuan dan Aplikasi Teknik Informatika*, vol. 12, no. 1, pp. 1–5, 2020.

[7] F. Marisa, A. R. Wardhani, W. Purnomowati, A. V. Vitianingsih, A. L. Maukar, and E. W. Puspitarini, "POTENTIAL CUSTOMER ANALYSIS USING K-MEANS WITH ELBOW METHOD," *JIKO (Jurnal Informatika dan Komputer)*, vol. 7, no. 2, pp. 307–312, 2023.

[8] E. F. A. Pratama, K. Khairil, and J. Jumadi, "Implementasi Metode K-Means Clustering Pada Segmentasi Citra Digital," *Jurnal Media Infotama*, vol. 18, no. 2, pp. 291–301, 2022.

[9] H. Kurniawan and S. Defit, "Data Mining Menggunakan Metode K-Means Clustering Untuk Menentukan Besaran Uang Kuliah Tunggal," *Journal of Applied Computer Science and Technology*, vol. 1, no. 2, pp. 80–89, 2020.

[10] B. A. B. II, "Tinjauan Pustaka dan Landasan Teori," *Dalam bab ini brisikan tentang tinjauan pustaka dan landasan teori sebagai pendukung yang digunakan untuk pembahasan tentang pembuatan dan cara kerja dari aplikasi Sistem Hitung Cepat Hasil Pemilihan Kepala Daerah*, 2017.

[11] S. Riyadi, "Sistem Pengembangan Deteksi Kanker Prostat Berbasis Image Processing dengan Metode Convolutional Neural Network," *Explore IT: Jurnal Keilmuan dan Aplikasi Teknik Informatika*, vol. 14, no. 2, pp. 52–63, 2022.

[12] J. Li, Y. Cai, Q. Li, M. Kou, and T. Zhang, "A review of remote sensing image segmentation by deep learning methods," *Int J Digit Earth*, vol. 17, no. 1, p. 2328827, 2024.

[13] S. Mulyadi, F. Insani, S. Agustian, and L. Afriyanti, "Pengelompokan Data Pendistribusian Listrik Menggunakan Algoritma Mini Batch K-Means Clustering: Grouping Electricity Distribution Data Using The Mini Batch K-Means Clustering Algorithm," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 3, pp. 1051–1062, 2024.

[14] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Trans Big Data*, vol. 7, no. 3, pp. 535–547, 2019.

[15] C. Foltz, N. Schneider, B. Kausch, M. Wolf, C. Schlick, and H. Luczak, "Usability engineering," in *Collaborative and Distributed Chemical Engineering. From Understanding to Substantial Design Process Support: Results of the IMPROVE Project*, Springer, 2008, pp. 527–554.

[16] R. J. H. Butar-Butar and N. L. Marpaung, "Deep Learning untuk Identifikasi Daun Tanaman Obat Menggunakan Transfer Learning MobileNetV2," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 8, no. 2, pp. 142–148, 2023.

[17] S. Mittal, P. Rajput, and S. Subramoney, "A survey of deep learning on cpus: opportunities and co-optimizations," *IEEE Trans Neural Netw Learn Syst*, vol. 33, no. 10, pp. 5095–5115, 2021.

[18] N. P. Jouppi *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *Proceedings of the 44th annual international symposium on computer architecture*, 2017, pp. 1–12.