Modified MobileNet-V2 Convolution Neural Network (CNN) for Character Identification of Surakarta Shadow Puppets

Achmad Solichin*, Dwi Pebrianti, Painem, Sanding Riyanto
Faculty of Information Technology, Universitas Budi Luhur, Jakarta, Indonesia

***Abstract*—Shadow puppets or in Indonesian called as "wayang kulit" is one of Indonesia's native traditional arts that still exists to this day. This art form has been recognised by UNESCO since 2003. Wayang kulit is not just ordinary entertainment. It carries profound moral values, but is gradually being forgotten by the younger generation. To facilitate the public in recognizing wayang kulit characters, a desktop-based application was developed using Canny edge detection for image extraction and a modified MobileNet-V2 CNN algorithm for character identification. The dataset used in this research was sourced from Google and Instagram, with 22 names of wayang kulit characters serving as classes. The identification results for 1,312 wayang kulit images (test data) using the classic CNN model yielded an accuracy of 50%, precision of 53%, and recall of 47%. Meanwhile, with the modified MobileNet-V2 CNN model, called custom CNN gives an accuracy of 92%, precision of 93%, and recall of 92%. From the result, it is shown that the custom CNN has high performance, where it has a few false positive predictions in detecting the characters of wayang kulit. Additionally, the result shows that the CNN model is robust and reliable for the task of identifying the wayang kulit characters. Based on the result, the model can be applied in preserving and promoting traditional wayang kulit art by helping to catalog and identify characters, making it more accessible to a wider audience, including the younger generation.**

***Keywords*—Wayang kulit; characters identification; Convolution Neural Network (CNN); machine learning; image processing**

I. Introduction

Wayang kulit is a form of puppet theatre from Central and East Java (Javanese culture) that utilizes figures made from buffalo or sometimes cowhide. Wayang performances can be shown with their shadows from behind the screen to reveal the intricacies of their details, but they can also be presented from the front to showcase the beauty of their artistry. The stories and narratives are drawn from the Ramayana and Mahabharata epics. In each story, there is typically a conflict between the virtuous (protagonist) and the villainous (antagonist) wayang characters.

Wayang kulit is one of Indonesia's native cultures, recognized by UNESCO as a Masterpiece of Oral and Intangible Heritage of Humanity, an awe-inspiring cultural narrative and beautiful cultural heritage, since November 7, 2003 [1]. This recognition serves as motivation and a call to all elements of the nation to continue preserving and safeguarding the essence of wayang kulit art. One way to do this is by striving to understand and recognize the characters in wayang kulit.

Sulaksono et al. mentioned that wayang kulit is footage or a representation of human life symbolized in shadows [2]. Wayang kulit can be used as a medium in the character teaching and learning [3]. Furthermore, it is believed that wayang is not only symbolized the human physical, but rather symbolized human nature [4]. By knowing and implementing the values brought by wayang kulit story, it is expected that the human life on earth will be peaceful and enriched with cultural wisdom.

However, most of the younger generation today are not familiar with the various types or names of wayang kulit characters, except for those born before the 1990s [5] [6] [7]. Furthermore, the various forms of wayang kulit characters can appear similar, making it challenging for the public to distinguish them.

While the COVID-19 pandemic has made wayang kulit performances available through YouTube videos, it only addresses the broadcast mechanism of these performances and does not solve the issue of the audience not recognizing the names of wayang kulit characters. This becomes a problem when an entire generation loses knowledge of these characters' names and identities.

Surakarta shadow puppets was most popular for its performance and craftsmanship; many people collected the wayang and it was widely spread throughout the globe [8]. Surakarta's wayang kulit becomes famous for several reasons which are the historical significance. Surakarta was the center of Javanese culture and art. Additionally, Surakarta is renowned for its high quality wayang kulit performance. Puppet makers and puppeteers in Surakarta are known for their exceptional craftmanship and skill in creating intricate shadow puppets. On top of that, Surakarta Sultanate has historically been a strong supporter of wayang kulit.

Each character in Surakarta's wayang kulit is dependent on its visual attribute. They may have unique facial features, clothing, and accessories. Color and attire are also significant attributes to identify each character in wayang kulit. Additionally, posture and gestures are other attributes that can be used for the identification of wayang kulit's character.

Arjuna is one of the characters in wayang kulit which belongs to Pandava brothers from the Indian epic, the Mahabharata. It is mentioned that Arjuna is representing humans with his heroic and noble characteristic [9]. Arjuna has a distinctive head with a crown or headgear, his facial features are often depicted with fine details, including eyes, nose and mouth.

Furthermore, Arjuna's body is usually slender and well-proportioned.

Research in identification of wayang kulit's characters was mainly involving the puppet experts, for example Ki M, Dim Hali Djarwosularso, Ki Manteb Soedharsono, Ki R. Ng. Soenarno Dutodiprojo, Ki Gaib Widopandoyo and Ki Sudirman Ronggodarsono [10]. Furthermore, there are some researchers in social studies tried to compile the visual of *wayang kulit*'s character into recorded documentation [11], [12].

With the advancement of Artificial Intelligence nowadays, it can be implemented in the identification of wayang kulit's characters [12]. This will lead to the preservation of wayang kulit as a cultural artifact. In the past five years, a substantial number of researchers have exhibited a keen interest in the application of Artificial Intelligence (AI) and Machine Learning (ML) for the identification of shadow puppet characters. The primary objective, naturally, is the preservation of this globally recognized cultural heritage.

A technique based on the Single Shot Multiple Detector (SSD) was developed to detect four Punakawan characters, achieving an accuracy of 98% [13]. Nevertheless, the method was limited to characters with clearly distinguishable visual traits. Later, the Convolutional Neural Network (CNN) approach was applied to classify five wayang characters using the Raspberry Pi 4 platform, reaching an accuracy between 96% and 97% [14]. However, this study only focused on a small subset of wayang characters and did not provide detailed insights into the implementation process on the Raspberry Pi.

Within the same timeframe, several studies also investigated the classification or recognition of wayang characters using a restricted number of character classes. One such approach involved the use of Support Vector Machine (SVM) combined with Gray Level Co-occurrence Matrix (GLCM) features to identify a set of five selected wayang figures. [15]. Simultaneously, another study embarked on a similar exploration by employing GLCM features but applied the Multi-Layer Perceptron (MLP) classification method to identify five distinct wayang characters [16]. Notably, both investigations yielded accuracy values that remained relatively modest.

In the continued exploration of wayang character recognition, subsequent studies have revisited and expanded upon earlier methodologies. A study conducted in 2021 re-applied the Support Vector Machine (SVM) classifier alongside Gray Level Co-occurrence Matrix (GLCM) features to classify the same five wayang characters as in prior research. However, the results did not indicate a significant improvement in classification accuracy compared to earlier findings [15].

Building upon previous efforts, another study introduced a novel two-stage approach [17]. The first stage involved the identification of six wayang characters, followed by a second stage in which web scraping techniques were employed to retrieve relevant textual information from online sources based on the identified characters. This study utilized the VGG16 deep learning architecture and reported an accuracy of 89%. An enhancement to this approach was later introduced using the Mask Region-based Convolutional Neural Network (Mask R-CNN) model [18], applied to the same dataset and set of characters. The adoption of Mask R-CNN led to an improved classification accuracy of 92%, indicating its potential superiority in handling object detection tasks involving complex traditional imagery.

In the past two years, there has been a marked shift towards the use of deep learning techniques for wayang character classification. One such study employed Convolutional Neural Networks (CNN) to categorize a substantial dataset of 430 wayang images into four distinct character groups [19], achieving an accuracy of 93%. A similar effort utilized CNN to classify 100 monochromatic wayang images into binary classes, distinguishing between protagonist and antagonist roles [20]. Recent research combines CNN and Hyperparameters tuning methods to identify five wayang characters [21]. However, the accuracy is still relatively low.

Another significant contribution explored the classification of 400 wayang images divided into four categories [22]. This study integrated a series of image preprocessing techniques, including Contrast Limited Adaptive Histogram Equalization (CLAHE), RGB color space transformations, Gaussian filtering, and thresholding. These methods culminated in the highest reported classification accuracy to date, reaching 98.75%.

In contrast to the prevailing trend of deep learning adoption, a more recent study employed the Extreme Learning Machine (ELM) algorithm combined with morphological feature extraction to identify five wayang characters [7]. While the methodology introduced an alternative perspective, its accuracy performance remained below the benchmarks established by deep learning-based approaches.

A comprehensive review of research pertaining to the identification of wayang characters conducted within a period of five years, has led to several research gaps. Firstly, most of the researches remain confined to a restricted wayang kulit's characters, notably favoring the more widely recognized and celebrated figures such as Pandawa [14], [17], [18], [19], Punawakan [13], [22], and a select cohort of other prominent characters. The preference for a limited subset of characters underscores a gap in the exploration of the broader spectrum of wayang figures.

Secondly, the efficacy of the methods employed has yet to attain an optimal level of performance. Notably, the deployment of Deep Learning methodologies generally affords superior accuracy in contrast to classical Machine Learning approaches. Nevertheless, further refinement and optimization of these Deep Learning techniques remain requisite to enhance their performance.

Thirdly, most of researchers are not considering the uniqueness of each wayang kulit's character within diverse regions of Indonesia. For instance, the Surakarta style wayang exhibits a distinct morphology characterized by its slender attributes, in stark contrast to the more robust Jogjakarta style. The intrinsic regional variations in wayang representations remain an underexplored facet within the contemporary research landscape, warranting more extensive investigation.

Based on the above description, the author proposes an alternative solution to create an application system capable of identifying the names of wayang kulit characters. This

application system adopts one branch of Artificial Intelligence, namely Image Processing.

In this study, wayang kulit's characters identification using Canny edge detection and Convolution Neural Network (CNN) is proposed. The paper will be divided into 4 sections. The first section is the introduction and the motivation of conducting the research. The second section will discuss the research methodology including the proposed technique. The proposed technique will be combining Canny edge detection and CNN. The third section will be the result and discussion. In this section, quantitative and qualitative analysis will be discussed in depth. Last section will be the conclusion and future works.

II. METHODOLOGY

The study will be started with the data collection and data pre-processing, continued with design of Convolution Neural Network (CNN) model for wayang kulit's characters identification and the analysis method.

A. Data Collection and Pre-Processing

The dataset used in this research is obtained from photos of wayang kulit on Google and several Instagram accounts of wayang kulit craftsmen. There are a total of 22 characters represented in the folder, with the number of photos (wayang kulit) amounting to 6,576, which resulted from augmenting the initial set of 411 images with a .jpg extension. Fig. 1 shows an example from the dataset.

Fig. 1. Image data set of wayang kulit.

The names of the shadow puppet (wayang kulit) characters used in this dataset are characters from the Mahabharata story, including Abimanyu, Anoman (Hanuman), Arjuna, Bagong, Baladewa, Bima (Bhima), Buta, Cakil, Durna (Drona), Dursasana, Duryudana, Gareng, Gatotkaca, Karna, Kresna (Krishna), Nakula Sadewa, Patih Sabrang, Petruk, Puntadewa, Semar, Sengkuni, and Togog.

The dataset consists of 411 wayang kulit image files collected during the data collection phase, with various dimensions and in .jpg format. These images will be further processed to enhance their variety using augmentation techniques and preprocessing methods.

In the next stage, several processes are carried out to increase the dataset's variety using data augmentation techniques and edge detection. Data augmentation involves modifying an existing dataset to make it more diverse. Below are the details of the pre-processing stage for shadow puppet images.

- Flip Left-Right: A process of flipping an image horizontally or vice versa. The flip process percentage is 100%, meaning that if there are five images in a folder, all five of them will be flipped.
- Flip Up-Down: A process of flipping an image vertically or vice versa. The flip process percentage is 100%, the same case as Flip Left-Right, when there are 5 images in a folder, all five of them will be flipped.
- Rotation: This process involves rotating an image by a certain angle. In this study, the angle used is 25 degrees.
- Zooming or Scaling: A process of enlarging or reducing an image. In this study, zooming is performed at percentages of 40%, 100%, 80%, and 120% for each axis.
- Edge Detection: The processing technique that is used to identify the boundaries (edges) of objects or regions within an image. In this study, the edge detection function used is Canny from the OpenCV library.

Before the data splitting process, the shadow puppet images resulting from pre-processing will be grouped into their respective 22 folders, according to the names of the shadow puppet characters. After data grouping, the dataset will be divided into two parts: test data and training data, with a ratio of 20:80.

In both the training and test datasets, there are 22 folders representing labels for shadow puppet characters. Each folder contains shadow puppet images that have undergone augmentation and preprocessing.

B. Design of Convolution Neural Network

MobileNet-v2 is chosen as the base model for the wayang kulit's characters identification due to several advantages as listed below.

- Efficiency: MobileNet-V2 is known for its high computational efficiency and low memory footprint. It is optimized for mobile devices, making it suitable for real-time applications where computational resources are limited [23].
- Speed: Its lightweight architecture allows for rapid inference, essential for real-time applications. Recent evaluations in medical image classification confirm that MobileNet-V2 achieves low latency while maintaining reliable performance [24].
- Accuracy: While MobileNet-V2 may not be as accurate as some larger and more complex models, it still provides competitive accuracy for various image recognition tasks. Its balance between speed and accuracy makes it a popular choice for many real-world applications [25], [26].
- Transfer Learning: MobileNet-V2 is often used as a base model for transfer learning. Transfer learning involves fine-tuning a pre-trained MobileNet-V2 on a specific dataset, which can yield excellent results with relatively little training data [24], [27].
- Small Model Size: MobileNet-V2 models have a smaller file size compared to many other deep learning models. This is beneficial for mobile applications where storage space is limited [28].
- Low Power Consumption: MobileNet-V2's efficiency extends to power consumption [29], making it suitable for battery-powered devices like smartphones and drones.
- Versatility: MobileNet-V2 can be used in a wide range of computer vision tasks, including object detection, image classification, and image segmentation, making it a versatile choice for developers.

The contribution of the study will be on the modification of MobileNet-v2 model which is done by removing the last two layers of the original model. There are several hypotheses in removing the last two layers of the original model. The first one is retaining the feature obtained from the feature extraction layers. The last two layers are the classification purpose layer. By removing these two last layers, the CNN model tends to keep the features extracted from the previous layers. The second hypothesis is the proposed model will speed up the training process. As the layers become fewer, the training process will be conducted faster than the original model. The next hypothesis will be the ability to add custom output layers. This custom output layers can be adjusted to be appropriate for certain cases, for example object localization tasks, a multi-label classification layer for multiple object recognition which is the main objective in this study etc.

1) Classic Model of Convolution Neural Network (CNN). The modeling stage is carried out to extract features from the preprocessed wayang kulit images using the Convolutional Neural Network (CNN) architecture. In the CNN architecture, there are several main processes known as layers, including the Convolutional Layer, Pooling Layer, Flatten Layer, Dense Layer, and Activation. Fig. 2 presents an illustration of the CNN architecture as follows.

The first step in image processing using the CNN algorithm is the convolution operation. An image is represented as a matrix containing pixel values. In the first layer, which is convolution layer, convolution is performed between the matrix of the input image and a kernel or filter with a specific matrix order and values. The result of the image convolution process is called a feature map, and there are four (4) of them (because there are 4 kernels).

The second step is pooling or subsampling, which serves to reduce the dimensions of each feature map resulting from the convolution operation. At this stage, activation processes are typically applied. In this study, ReLU activation is used because it speeds up the training process compared to other activation functions (such as sigmoid, tanh, linear, etc.) [8].

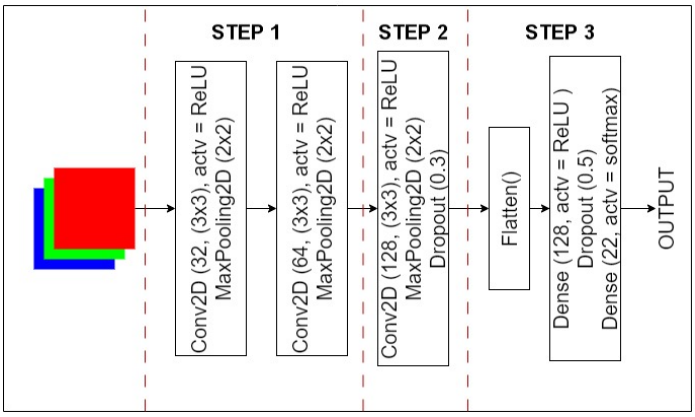

Fig. 2. Classical model of convolution neural network.

The next step is flattening, which is the process of transforming the feature maps into a one-dimensional array that will then be fed into the Neural Network layer (Dense). Subsequently, this array becomes the input to the neural network layer. The final layer is the Dense or Fully Connected Layer, with the number of nodes representing the number of labels or classes. In cases where there is only one output node, the sigmoid activation function is used. However, in this research, since there is more than one label, the softmax activation function is employed, which is well-suited for multiclass classification.

2) MobileNet-v2 Model. The original MobileNet-v2 model is shown in Fig. 3. In essence, MobileNet-V2 performs the same operations as a basic CNN architecture in its layers. However, the key difference lies in the greater number and complexity of these layers. A detailed illustration of the MobileNet-V2 architecture is presented in Fig. 4.

In the MobileNet-V2 architecture, there are two (2) types of blocks: residual blocks (stride = 1) and other blocks with (s = 2). Each of these two blocks consists of three (3) layers, including: 1x1 convolution with ReLU6, depth-wise convolution, and 1x1 convolution without any non-linearity. Detailed information regarding these blocks in the MobileNet-V2 architecture is presented in Table I.

In this study, the value of t is set to be 6 for the ReLU6 for all of the main experiment. Therefore, if the input has 64 channels, then it will result an output dimension $64 \times t$ or $64 \times 6 = 384$ channels.

In Table II, the bottleneck section contains the input and output between the model, while the inner layers encapsulate the model's ability to transform input from lower-level concepts (i.e., pixels) into higher-level descriptors (i.e., image categories). Ultimately, similar to the residual connections in traditional CNNs, shortcuts between bottlenecks enable faster training and improved accuracy.

3) Modified MobileNet-v2. Based on the descriptions of the CNN and MobileNet-V2 architectures mentioned earlier, in this research, the author combines both of them. Modifications to the MobileNet-V2 model were made because it did not align with the needs of this research problem, which is to classify 22 names of wayang kulit's characters. In the default MobileNet-V2, the last layer has 1000 nodes (intended for classifying 1000 types of images). Therefore, modifications were necessary by removing some layers and adding them as needed.

TABLE I. DETAIL OF ARCHITECTURE BLOCK OF MOBILENET-V2

Input	Operator	Output
$h \times w \times k$	$1 \times 1\ conv2d, ReLU6$	$h \times w \times tk$
$h \times w \times tk$	$3 \times 3\ dwise, ReLU6$	$\frac{h}{s} \times \frac{w}{s} \times tk$
$\frac{h}{s} \times \frac{w}{s} \times tk$	$Linear\ 1 \times 1\ conv2d$	$\frac{h}{s} \times \frac{w}{s} \times k'$

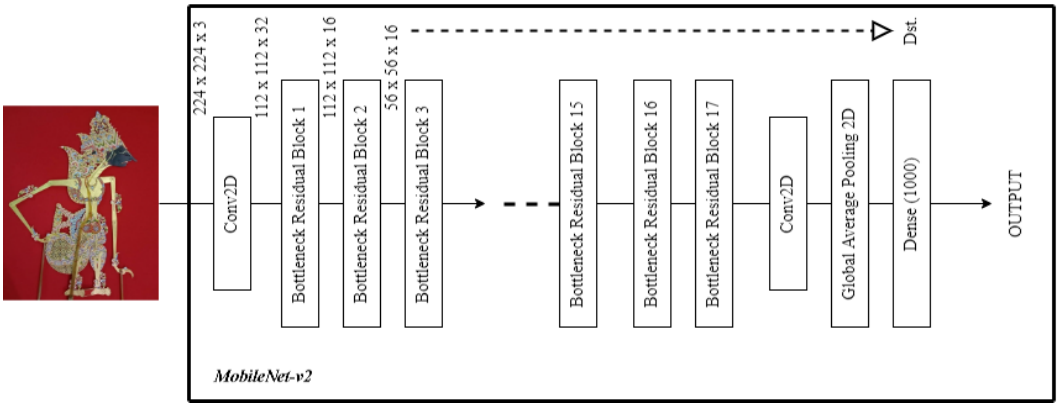

Fig. 3. Architecture of original MobileNet-V2.

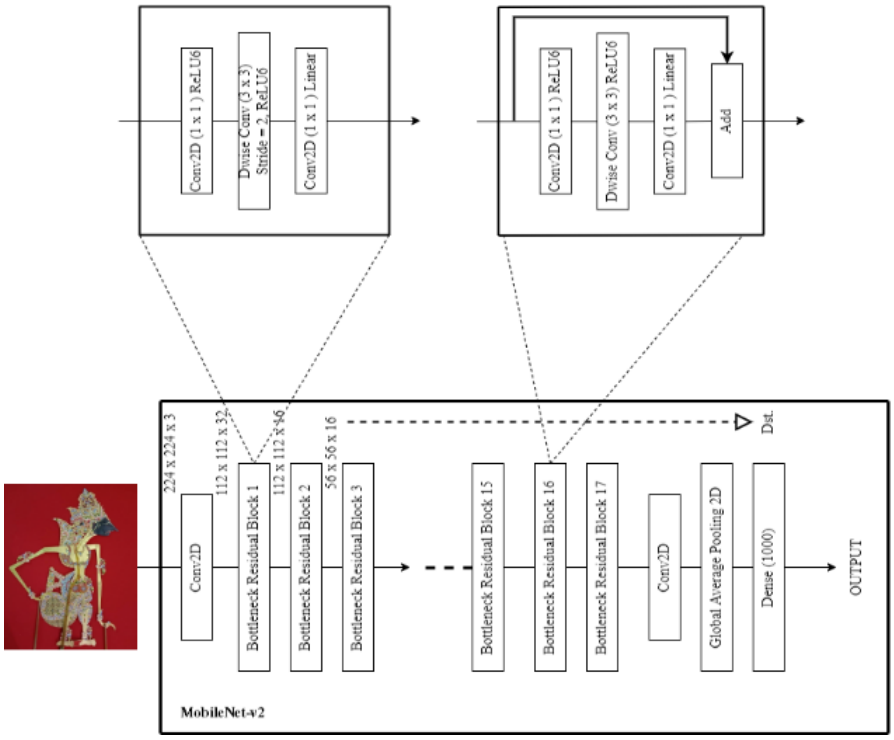

Fig. 4. Architecture of modified MobileNet V2.

TABLE II. INPUT AND OUTPUT MODEL (BOTTLENECK SECTION)

Input	Operator	*t*	*c*	*n*	*s*
$224^2 \times 3$	*Conv2d*	-	32	1	2
$112^2 \times 32$	*bottleneck*	6	16	1	1
$112^2 \times 16$	*bottleneck*	6	24	2	2
$56^2 \times 24$	*bottleneck*	6	32	3	2
$28^2 \times 32$	*bottleneck*	6	64	4	2
$14^2 \times 64$	*bottleneck*	6	96	3	1
$14^2 \times 96$	*bottleneck*	6	160	3	2
$7^2 \times 160$	*bottleneck*	6	320	1	1
$7^2 \times 320$	*Conv2d 1x1*	-	1280	1	1
$7^2 \times 1280$	*Agypool 7x7*	-	0	1	-
$1 \times 1 \times 1280$	*Conv2d 1x1*	-	K	-	

In our design, the new layers—comprising a dropout and dense classification head—are inserted after the feature extractor (see Fig. 5) and before the classification head. This position was chosen based on standard transfer learning practices, which leverage the pretrained backbone for general feature extraction and adapt the final layers to the specific target domain [24]. By removing the original 1000-class classifier and inserting task-specific layers at this juncture, we retain the rich visual features learned from large-scale datasets (e.g., ImageNet) while optimizing the model for 22-character classification in our domain. This approach is well-supported in recent lightweight CNN studies, where modifying only the head allows for efficient adaptation with minimal retraining [23].

The last two layers in the default MobileNet-V2 architecture were removed using the "include_top = False" command. By doing this, additional layers based on the basic CNN concept can be added. In this research, the author added Rescaling, Dropout, and Dense (22) layers, as shown in Fig. 5 as the modification to the original MobileNet-V2 model.

Several things to consider in the custom model training process are the number of training and testing data, epochs (iterations), and the validation loss-validation accuracy values. Models with good accuracy will be exported with a .h5 file extension.

C. Test Data Identification

In the test data identification phase, testing is performed on the custom CNN model which combined the MobileNet-V2 base model with new proposed layers. The test data consists of raw wayang kulit images, before pre-processing.

Each of wayang kulit's characters is labeled and put into an array. Once the matrix of image input is obtained, then the convolution process will be conducted.

The result of this identification is the name of the wayang kulit character along with a similarity percentage. In this research, the output of the identification falls under the category of multiclass classification, as it involves labels of more than two classes.

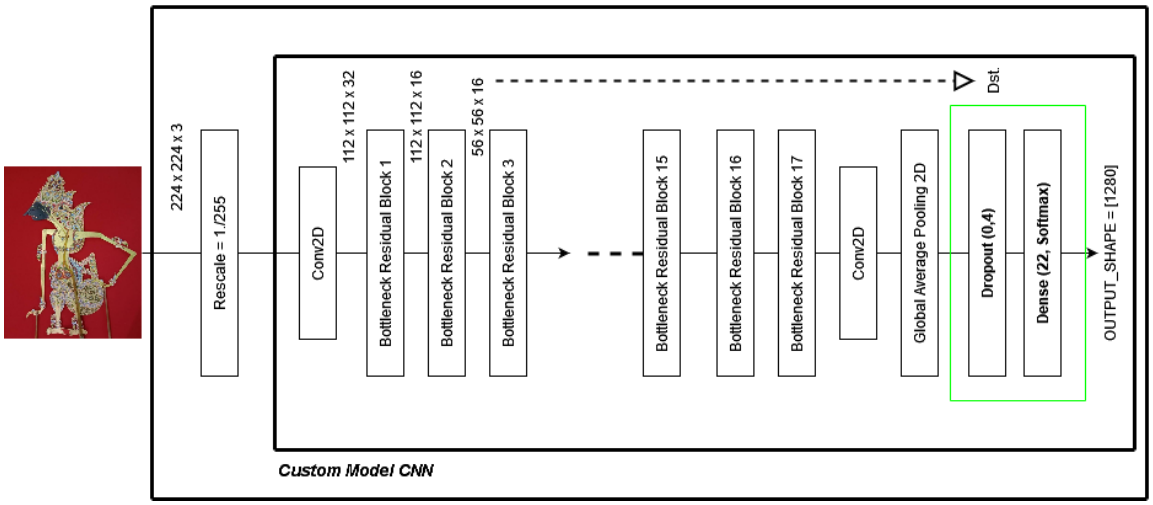

Fig. 5. Modified MobileNet-v2 model.

D. Performance Analysis Method

The performance analysis is conducted by measuring the accuracy, precision, and recall values of the trained model using the proposed algorithm. In this research, testing is done by comparing several predicted data which is the results of the classification phase with a set of actual data, results of the labeling phase. The term "several predicted data" refers to a set of data processed through the CNN algorithm, pre-trained model.

Accuracy is the degree of closeness between predicted values and actual values as shown in Eq. (1).

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

Precision is the level of accuracy in providing requested information compared to the responses given by the system as shown in Eq. (2).

$$Precision = \frac{TP}{(TP+FP)} \quad (2)$$

Recall is the system's success rate in rediscovering specific information as shown in Eq. (3).

$$Recall = \frac{TP}{(TP+FN)} \quad (3)$$

where

- TP: True Positive represents data labeled as X and predicted as X. For example, test data 1 labeled as "bagong" and predicted as "bagong."
- TN: True Negative represents data labeled as other than X and predicted as other than X. For example, test data 1 labeled as something other than "bagong" and predicted as something other than "bagong."
- FP: False Positive represents data labeled as NOT X but predicted as X. For example, data 1 is labeled as something other than "bagong" but predicted as "bagong."
- FN: False Negative represents data labeled as X but predicted as other than X. For example, test data 1 is labeled as "bagong" but predicted as something other than "bagong."

III. Results and Discussion

This section will discuss the result obtained in the study. The discussion will start with the data pre-processing, the performance of original model of MobileNet-v2 and the modified one and lastly the visualization of the result.

A. Data Collection and Data Pre-Processing

As mentioned in Section II (A), the total number of raw data obtained from Google and Instagram from duration 18 April to 18 May 2022 was 411 images. All images are saved in .jpg format.

These 411 images were then undergone a set of operations in order to increase the total number of images used as the training data set. At the final stage of the data collection, a total of 6,576 images were produced and used as the data set.

1) Flip Left-Right. Fig. 6 shows a snippet of the dataset after being processed with the Flip Left-Right method. The wayang kulit images, which originally faced left as shown in index (a), are flipped to face right as in index (b), and vice versa. In this step, the total number of the images data becomes 822, where 411 images are the original images, and another 411 images are the left-right flipped images.

Fig. 6. Image result from flip left right process.

2) Flip Up-Down. The second step to generate wayang kulit's images for the data set is by flipping up-down the 822 images obtained in Section III(A)(1). The sample result is shown in Fig. 7. As seen in the figure, only the orientation of the images is flipped from up to down and vice versa. By doing this process, the total number of images will be increasing double, which is from 822 to 1,644 images.

(a) (b)

Fig. 7. Image result from flip up down process.

3) Rotation. The third step for generating image data set is to do rotation to the data set obtained in Section III(A)(2). Generally, the rotation can be done in any values of angle of rotation. In this study, after conducting several experiments, the numbers of +25 and -25 are selected to be the best values for conducting the rotation to the images. The strongest reason for this choice is the estimated range of the shadow puppet's tilt when a puppeteer performs with a puppet. The reference axis is the y-axis or can be considered as being at the center point of the image. Fig. 8 shows an example of images under the rotation process. The total number of wayang kulit's images obtained in Section III(A)(2) is 1,644 images. After the rotation process, 3,288 images data are now available for the training and validation data set.

(a) (b)

Fig. 8. Image result from rotation process.

4) Zooming. The last process conducted in the research to increase the total number of images in the data set is zooming process. The purpose of the zooming process is to introduce variations in the scale or size of the input images. Some advantages of conducting zoom process are improved generalization, avoid overfitting, increase the accuracy of the model since more data is provided and improve the robustness of the model.

When applying zoom augmentation, balance must be considered. Too much zooming can distort images to the point where they no longer resemble the original object, making it harder for the model to learn. Therefore, the degree of zoom should be chosen carefully based on the specific problem and dataset. In this study, combination of (40%, 100%) for scale X, and (80%, 120%) for scale Y are used. This means that the system will generate a set of new image data set with the scaling range between 40% to 100% for the width of the image and range between 80% to 120% for the height of the image.

(a) (b)

Fig. 9. Image result from zooming process.

Fig. 9 shows the example result of images after the zooming process. As seen in the figure, the images have some deformation becoming bigger or smaller compared to the original images. After the zooming process, the 3,288 of total images number obtained from the previous process now becomes 6,576 images. This total number of images is sufficient for the CNN model to obtain a good performance.

5) Edge detection. The edge detection is conducted for the purpose of feature extraction. Canny detection is used in this study for the reasons of better accuracy result and also faster processing time in edge detection compared to other methods. Especially for the images of wayang kulit which are in 2 dimensions form.

(a) (b)

Fig. 10. Image result from edge detection process using canny algorithm.

Fig. 10 shows the result of *wayang kulit*'s images after the edge detection process using Canny algorithm. As can be seen from the figure, the edge of each character in the original image can be detected without any loss of information. This result will improve the performance of the CNN model in identifying the character of each *wayang kulit*'s image.

B. Comparison of CNN Models

This section will discuss the training result of *wayang kulit*'s character by using classic model of Convolution Neural Network (CNN) and modified MobileNet-V2 CNN model. The

performance evaluation will be focusing on the parameters of accuracy, precision, and recall.

1) Classic model. As explained in section II.B.1, classic CNN is used as benchmark for the proposed algorithm which is modified MobileNet-V2.

Wayang kulit images used in this study has dimension of $224 \times 224 \times 3$ (RGB) . The classic CNN will have 3 convolution blocks. The first block will use 32 filter kernels, block 2 uese 64 filter kernels and block 3 uses 128 filter kernels. All of the block use Rectified Linier Unit (ReLU) as the activation function. Meanwhile, 2-dimension Maxpooling (MaxPooling2D) is used to generate feature maps.

At block 1, the value of 32 for the filter kernels was chosen because the input for the block is the edge image of wayang kulit which is obtained from the pre-processed data. Edge image is low-level feature, hence 32 is the appropriate number for the process.

At the block 3, feature maps generated from the process will have dimension of $26 \times 26 \times 128$. After the convolution process, then the flatten process is conducted where the result is a 1-dimensional array that contains $26 \times 26 \times 128 = 86{,}528$ data.

The process above was conducted for all of the wayang kulit image data set. Each the image will be labeled automatically where the data set will be separated into 22 labels which represent the 22 wayang kulit characters.

The final step is training the model with the dataset that was previously divided, as described in subsection III(B)(3). In this model training, we conducted a total of 20 iterations (epochs) with an estimated time duration (ETA) of 45 minutes using Google Colaboratory. Table III shows the results of the training for the classic CNN model experiment. Referring to the result in the table, the model was considered to be overfitting. This is proven by the accuracy parameter where the value is 0.9308 or 93.08%. Overfitting happened due to several factors which are having too few layers, not proportional to the large amount of training data, which thousands in numbers or having poor dataset variation, tending to be similar. The model from the first experiment can still be used for identifying the test data. However, considering its low accuracy and the occurrence of overfitting, the modified MobileNet-V2 is then explored to perform the task of *wayang kulit*'s characters identification (as seen in Table IV).

2) Modified MobileNet-v2. As mentioned in Section II (B) (2), a modified version of MobileNet-V2 is proposed in this study for identifying the wayang kulit's characteristic.

TABLE III. PERFORMANCE OF CLASSIC CNN

Parameters	Results
Loss	0.1998
Accuracy	0.9308
Validation Loss	2.4045
Validation Accuracy	0.4958

TABLE IV. VALIDATION RESULT USING CLASSIC CNN

Label	*Precision*	*Recall*	*F1-score*	*Support*
Abimanyu	0,65	0,40	0,49	50
Anoman	0,58	0,60	0,59	70
Arjuna	0,56	0,63	0,59	79
Bagong	0,38	0,43	0,40	76
Baladewa	**0,34**	0,54	0,42	78
Bima	0,53	0,50	0,52	68
Buta	0,69	0,68	0,68	78
Cakil	0,60	0,70	0,65	60
Durna	0,50	0,26	0,34	38
Dursasana	0,55	0,48	0,51	58
Duryudana	0,42	0,38	0,40	60
Gareng	0,53	0,26	0,35	38
Gatotkaca	0,44	0,42	0,43	64
Karna	0,49	0,42	0,45	50
Kresna	0,44	0,65	0,53	74
Nakula_ Sadewa	0,62	0,39	0,48	38
Patih_ Sabrang	0,42	0,41	0,41	54
Petruk	0,41	0,35	0,38	65
Puntadewa	0,62	0,48	0,54	54
Semar	0,44	0,69	0,53	70
Sengkuni	0,51	0,38	0,44	52
Togog	**0,88**	0,37	0,52	38
Accuracy			0,50	1.312
Macro avg	0,53	0,47	0,49	1.312
Weighted avg	0,52	0,50	0,49	1.312

Basically, the modelling was carried out by utilizing the MobileNet-v2 base model. This model has undergone multiple rounds of training, making it already "smart". By utilizing ImageNet 1000 Class List, MobileNet-v2 is capable of classifying 1000 types of images because its output layer has 1,000 nodes. However, since wayang kulit images are not included in ImageNet, MobileNet-V2 is not capable of performing the task of identifying the characteristic of wayang kulit. To overcome the problem of the original MobileNet-v2 model, this study proposes a modified version of MobileNet-v2, which is explained in Section II (B) (3).

In this modeling process, the author added one layer before the MobileNet-v2 base model and several layers at the end. The added layers to the MobileNet-v2 base model include rescaling, dropout layers, and a dense (output) layer.

The rescaling process converts pixel values from the range of [0, 255] to the range of [0, 1]. This process is also known as input normalization. Scaling each image to the same range [0, 1] ensures that each image contributes more evenly to the total loss. This normalization makes it more likely for the neural network

to converge because it keeps coefficients within the range [0, 1] rather than [0, 255], which helps the model process input faster.

The next layer added to the MobileNet-v2 base model is the dropout layer. As explained in the result of classic CNN, dropout layer prevents model from overfitting. The dropout layer's dropout rate used here is 0.4 or 40%. This value is the best value obtained from several experimental results.

In the output layer, the dense layer is tailored to the number of labels or classes in this study, which totals 22 classes, using the softmax activation function. The output shape is also initialized to 1280, as shown in Table V.

The working mechanism of the proposed method which is modified MobileNet-V2 is fundamentally the same as the CNN architecture in Section III (B) (1). It includes convolution layers, pooling layers, flattening, dropout, and so on.

For the analysis purpose, two different experimental results by using the modified MobileNet-V2 were conducted. The first experiment is by using 20 iterations, while the second one is 100 iterations. The performance comparison of both experiments is shown in Table VI.

From the result it is shown that experiment with 20 iterations (epochs) needs shorter processing time, 25 minutes compared to 100 iterations that needs 30 minutes. By looking at the processing time itself, 20 iterations are better than 100 iterations due to the shorter time taken for the processing. However, this result cannot be used for judging the performance of a CNN model.

Other parameters to evaluate the performance of CNN models are listed in Table V. The comparison results shows that modified MobileNet-V2 with 100 iterations show better performance than model with 20 iterations. Overall, it is seen from the accuracy achieved, where for 20 and 100 iterations the accuracy is 78% and 82%, respectively. By giving a setting value100 for the iterations, the proposed model performance is 4% increased.

TABLE V. PARAMETER SETTING FOR MODIFIED MOBILENET-V2

Layer (Type)	Output Shape	Parameters #
Keras_layer (KerasLayer)	(None, 1280)	2257984
dropout (Dropout)	(None, 1280)	0
dense (Dense)	(None, 2)	28182

TABLE VI. PERFORMANCE COMPARISON OF MODIFIED MOBILENET-V2 UNDER DIFFERENT CONDITION

Parameter	20 Iterations	100 Iterations
1) Loss	0.6874	0.5386*
2) Accuracy	0.7802	0.8216*
3) Validation Loss	0.6412	0.5828*
4) Validation Accuracy	0.8056	0.8140*

Notes: value with * shows the best performance

C. Discussion

In this section, the performance of the classic CNN and modified MobileNet-V2 will be validated. For both CNN models that have been trained as explained in Section III (B), a new set of wayang kulit images was introduced. The performance of each model will be analyzed in terms of loss, accuracy, validation accuracy and validation loss.

The test is conducted for all of 22 wayang kulit's characters. The identification result by using classic CNN is shown in Table VII. The table shows the result for each character being identified by the classic CNN model. The lowest precision is for 'Baladewa' character and the highest is for 'Togog' character. Both characters are shown in Fig. 11. As seen from Fig. 11, 'Baladewa' has more edges compared to 'Togog'. This proves that the more edges contained in an image the lower the accuracy achieved by the model.

TABLE VII. VALIDATION RESULT USING MODIFIED MOBILENET-V2

Label	*Precision*	*Recall*	*F1-score*	*Support*
Abimanyu	0,94	0,88	0,91	50
Anoman	0,93	0,97	0,95	70
Arjuna	0,97	0,99	0,98	79
Bagong	0,88	0,96	0,92	76
Baladewa	0,80	0,91	0,85	78
Bima	0,98	0,93	0,95	68
Buta	0,99	0,96	0,97	78
Cakil	0,93	0,95	0,94	60
Durna	**1,00**	0,97	0,99	38
Dursasana	0,93	0,88	0,90	58
Duryudana	0,96	0,78	0,86	60
Gareng	0,94	0,87	0,90	38
Gatotkaca	0,95	0,91	0,93	64
Karna	0,95	0,82	0,88	50
Kresna	0,82	0,97	0,89	74
Nakula_ Sadewa	**1,00**	0,87	0,93	38
Patih_ Sabrang	**0,78**	0,87	0,82	54
Petruk	0,97	0,92	0,94	65
Puntadewa	0,93	0,94	0,94	54
Semar	0,93	0,97	0,95	70
Sengkuni	0,94	0,92	0,93	52
Togog	0,95	0,92	0,93	38
Accuracy			0,92	1.312
Macro avg	0,93	0,92	0,92	1.312
Weighted avg	0,93	0,92	0,92	1.312

Meanwhile, the performance result for each character identification using the modified MobileNet-V2 is summarized in Table VIII. From the result, it is shown that the lowest precision is obtained from the 'Patih Sabrang' character identification, where the result is 78% precise. While, for the highest precision, there are two characters identified successfully which are 'Durna' and 'Nakula Sadewa' that achieve 100% precision.

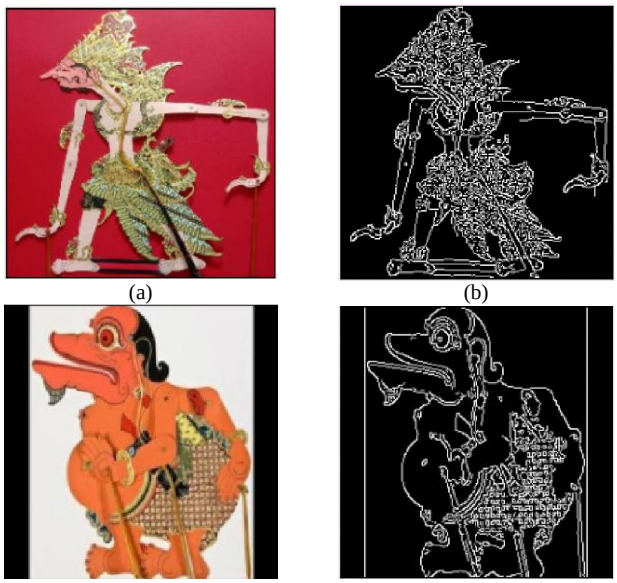

Fig. 11. (a) Baladewa raw image (b) Baladewa in edge image (c) Togog raw image (d) Togog in edge image.

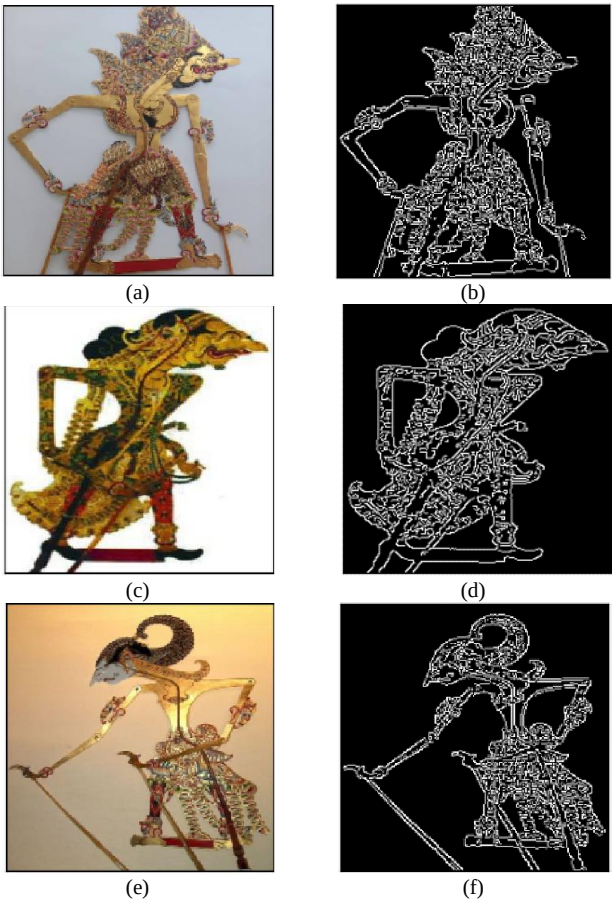

Fig. 12. (a) Patih Sabrang raw image (b) Patih Sabrang in edge image (c) Durna raw image (d) Durna in edge image (e) Nakula Sadewa raw image (f) Nakula Sadewa in edge image.

The image comparison between the three characters is shown in Fig. 12. As depicted in the picture, Patih Sabrang has the most edges as its feature. This makes the modified MobileNet-V2 has low accuracy in identifying the character, which is only 78%. However, this result is better compared to classic CNN which has 42% of accuracy. As the total number of edges in the image is decreasing, the proposed method has better performance in identifying the characters. This is proven by Durna and Nakula Sadewa characters identification since it achieves 100% of accuracy.

Table VIII shows the comparison between both models in terms of accuracy, precision, and recall. As seen from the table, the proposed method, modified MobileNet-V2 has the best result for all the performance analysis. This shows that the proposed method has potential to be implemented in real identification of *wayang kulit*'s characters.

TABLE VIII. PERFORMANCE COMPARISON OF VALIDATION PROCESS FOR CLASSIC CNN AND MODIFIED MOBILENET-V2

Parameters	Classic CNN (%)	Modified MobileNet-V2 (%)
Accuracy	50	**92**
Precision	53	**93**
Recall	47	**92**

Further analysis was conducted on the character of Baladewa. Classic CNN gave 34% accuracy in the identification of Baladewa's character. Meanwhile, the identification increased to 80% when using the proposed method. The result shows that in spite of the complexity of Baladewa's character which is shown by the large number of edges in the image, MobileNet-V2 successfully detected the character. This evidence demonstrates the outstanding performance of the proposed method.

IV. CONCLUSION

The goal of the study is to design an identification system for wayang kulit's characters by using Convolution Neural Network based model. There are several items that can be summarized from the study. The edge detection method using Canny Edge Detection was successfully applied for preprocessing the dataset of shadow puppet / wayang kulit images with low threshold value is 100 and high threshold value is 200.

Two different Convolution Neural Network (CNN) namely classic CNN and modified MobileNet-V2 were successfully designed. The processing time during the training process was 25 and 30 minutes for classic CNN and modified MobileNet-V2, respectively. The performance of both models in terms of validation loss is 2.4045 and 0.5828 for classic CNN and proposed model, respectively. Where, the validation accuracy shows that modified MobileNet-V2 has better performance with 0.8140 or 81% compared to classic CNN that achieved only 0.4958 or 50%. By looking at the training performance, it is seen that the proposed model has better performance compared to classic CNN for identifying the wayang kulit characters.

During the validation process, where 1,312 new image data set were introduced to the models, the performance of both models in terms of accuracy, precision and recall shows that the proposed model has better results. Modified MobileNet-V2 has an accuracy of 50%, precision of 53%, and recall of 47%. Meanwhile, the proposed model has an accuracy of 92%, precision of 93%, and recall of 92%. Overall, the performance result shows that the proposed method, modified MobileNet-V2 has excellent performance in identifying the *wayang kulit*'s characters.

As for future works, authors suggest some items listed below for further exploration on the study.

- The system's development can be integrated with other methods to achieve better results. For example, the hyper-parameters of the CNN models are adjusted by using an optimization algorithm.
- The current system can only detect one character in a single scene. For further improvement, multiple character identification can be designed to make the system more effective.
- The labeling process conducted in this study is still in manual process. To save processing time, automatic labelling processes can be developed.

ACKNOWLEDGMENT

Thank you to Universitas Budi Luhur for fully supporting this research.

REFERENCES

[1] UNESCO, "Masterpieces of the Oral and Intangible Heritage of Humanity: Proclamations 2001, 2003 and 2005; 2006," 2006. Accessed: Sep. 21, 2023. [Online]. Available: https://unesdoc.unesco.org/ark:/48223/pf0000147344

[2] D. Sulaksono and K. Saddhono, "Ecological Concept of Wayang Stories and the Relation with Natural Conservation in Javanese Society," KnE Social Sciences, vol. 3, no. 9, p. 58, Jul. 2018, doi: 10.18502/kss.v3i9.2611.

[3] Sumpana, Sapriya, E. Malihah, and K. Kumalasari, "Wayang Kulit As A Medium Learning Character," in Proceedings of the International Conference Primary Education Research Pivotal Literature and Research UNNES 2018 (IC PEOPLE UNNES 2018), Paris, France: Atlantis Press, 2019. doi: 10.2991/icpeopleunnes-18.2019.12.

[4] M. Isa Pramana and W. Yudoseputro, "Unsur Tasawuf dalam Perupaan Wayang Kulit Purwa Cirebon dan Surakarta," 2007.

[5] D. A. Ghani, "Digital Puppetry: Comparative Visual Studies between Javanese & Malaysian Art," 2018. [Online]. Available: http://www.ripublication.com

[6] Y. S. Lim, "Wayang Kulit and Its Influence on Modern Entertainment." [Online]. Available: www.iafor.org

[7] F. Fatmayati, M. Nugraheni, R. Nuraini, and F. Rossi, "Classification of Character Types of Wayang Kulit Using Extreme Learning Machine Algorithm," Building of Informatics, Technology and Science (BITS), vol. 5, no. 1, Jun. 2023, doi: 10.47065/bits.v5i1.3568.

[8] A. Ahmadi, "Arts and Design Studies The Creativity of Wayang Kulit (Shadow Puppet) Crafts in Surakarta," Arts and Design Studies, vol. 58, 2017, [Online]. Available: www.iiste.org

[9] R. Kelkar, P. Mokracek, S. K. Nimbalkar, and N. Gandhi, "A Study Of Arjuna's Qualities And Their Implications In Today's Management Scenario." [Online]. Available: http://journalppw.com

[10] W. Haryana, J. Masunah, and T. Karyono, "Wanda Wayang Kulit Surakarta in Perspective Visual Communication Design," 2022. doi: 10.2991/assehr.k.220601.067.

[11] B. Grahita and T. Komma, "Identification of The Character Figures Visual Style in Wayang Beber of Pacitan Painting."

[12] N. Khairina, R. Karenina Isabella Barus, M. Ula, and I. Sahputra, "Preserving Cultural Heritage Through AI: Developing LeNet Architecture for Wayang Image Classification," IJACSA) International Journal of Advanced Computer Science and Applications, vol. 14, no. 9, pp. 174–181, 2023, [Online]. Available: www.ijacsa.thesai.org

[13] A. N. A. Thohari and R. Adhitama, "Real-Time Object Detection For Wayang Punakawan Identification Using Deep Learning," JURNAL INFOTEL, vol. 11, no. 4, Dec. 2019, doi: 10.20895/infotel.v11i4.455.

[14] K. Wisnudhanti and F. Candra, "Image Classification of Pandawa Figures Using Convolutional Neural Network on Raspberry Pi 4," in Journal of Physics: Conference Series, IOP Publishing Ltd, Nov. 2020. doi: 10.1088/1742-6596/1655/1/012103.

[15] M. Muhathir, M. H. Santoso, and D. A. Larasati, "Wayang Image Classification Using SVM Method and GLCM Feature Extraction," JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING, vol. 4, no. 2, pp. 373–382, Jan. 2021, doi: 10.31289/jite.v4i2.4524.

[16] M. H. Santoso, D. A. Larasati, and Muhathir, "Wayang Image Classification Using MLP Method and GLCM Feature Extraction," Journal of Computer Science, Information Technologi and Telecommunication Engineering, vol. 1, no. 2, pp. 111–119, Sep. 2020, doi: 10.30596/jcositte.v1i2.5131.

[17] I. B. K. Sudiatmika, Pranowo, and Suyoto, "Indonesian Traditional Shadow Puppet Image Classification: A Deep Learning Approach," in 2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE), IEEE, Jul. 2018, pp. 130–135. doi: 10.1109/ICITEED.2018.8534776.

[18] I. B. K. Sudiatmika, M. Artana, N. W. Utami, M. A. P. Putra, and E. G. A. Dewi, "Mask R-CNN for Indonesian Shadow Puppet Recognition and Classification," in Journal of Physics: Conference Series, IOP Publishing Ltd, Feb. 2021, pp. 1–8. doi: 10.1088/1742-6596/1783/1/012032.

[19] W. Supriyanti and A. Anggoro, "Classification of Pandavas Figure in Shadow Puppet Images using Convolutional Neural Networks," Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, vol. 7, no. 1, pp. 18–24, 2021, [Online]. Available: http://tokohwayangpurwa.

[20] A. P. Wibawa, W. A. Yudha Pratama, A. N. Handayani, and A. Ghosh, "Convolutional Neural Network (CNN) to determine the character of wayang kulit," International Journal of Visual and Performing Arts, vol. 3, no. 1, pp. 1–8, Jun. 2021, doi: 10.31763/viperarts.v3i1.373.

[21] P. Sugiartawan, P. W. Aditama, Welda, A. Y. Willdahlia, N. N. Dita Ardiani, and N. W. Wardani, "Optimation of Convolutional Neural Networks With Hyperparameter to Identification Indonesian Traditional Puppet," in 2024 IEEE International Symposium on Consumer Technology (ISCT), IEEE, Aug. 2024, pp. 198–202. doi: 10.1109/ISCT62336.2024.10791092.

[22] Kusrini, M. R. A. Yudianto, and H. Al Fatta, "The effect of Gaussian filter and data preprocessing on the classification of Punakawan puppet images with the convolutional neural network algorithm," International Journal of Electrical and Computer Engineering, vol. 12, no. 4, pp. 3752–3761, Aug. 2022, doi: 10.11591/ijece.v12i4.pp3752-3761.

[23] N. Isong, "Building Efficient Lightweight CNN Models," May 2025, doi: 10.48550/arXiv.2501.15547.

[24] Z. Gao, Y. Tian, S.-C. Lin, and J. Lin, "A CT Image Classification Network Framework for Lung Tumors Based on Pre-trained MobileNetV2 Model and Transfer learning, And Its Application and Market Analysis in the Medical Field," Applied and Computational Engineering, vol. 133, no. 1, pp. 90–96, Jan. 2025, doi: 10.54254/2755-2721/2025.20605.

[25] Q. DU, Z. LIU, Y. SONG, N. WANG, Z. JU, and S. GAO, "A Lightweight Dendritic ShuffleNet for Medical Image Classification," IEICE Trans Inf Syst, p. 2024EDP7059, 2025, doi: 10.1587/transinf.2024EDP7059.

[26] P. Shourie, V. Anand, D. Upadhyay, S. Devliyal, and S. Gupta, "Scalable Fire Classification with MobileNetV2-Driven Convolutional Neural Networks," in 2024 IEEE International Conference on Communication, Computing and Signal Processing (IICCCS), IEEE, Sep. 2024, pp. 1–5. doi: 10.1109/IICCCS61609.2024.10763572.

[27] H. Lokhande and S. R. Ganorkar, "Object detection in video surveillance using MobileNetV2 on resource-constrained low-power edge devices," Bulletin of Electrical Engineering and Informatics, vol. 14, no. 1, pp. 357–365, Feb. 2025, doi: 10.11591/eei.v14i1.8131.

[28] H. Wang et al., "Spectral Demodulation of Tapered Microfiber Grating Using MobileNet," IEEE Sens J, vol. 25, no. 2, pp. 2791–2797, Jan. 2025, doi: 10.1109/JSEN.2024.3507099.

[29] D. T. Speckhard, K. Misiunas, S. Perel, T. Zhu, S. Carlile, and M. Slaney, "Neural architecture search for energy-efficient always-on audio machine learning," Neural Comput Appl, vol. 35, no. 16, pp. 12133–12144, Jun. 2023, doi: 10.1007/s00521-023-08345-y.