Accredited Ranking SINTA 2
Decree of the Director General of Higher Education, Research and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online on: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 7 No. 6 (2023) 1394 - 1402 ISSN Media Electronic: 2580-0760

MobileNetV3-based Handwritten Chinese Recognition Towards the Effectiveness of Learning Hanzi

Suwarno[1], Tony Tan[2], Jonathan[3]
[1,2,3] Department of Information Systems, Faculty of Computer Science, Universitas Internasional Batam, Batam, Indonesia
[1]suwarno.liang@uib.ac.id, [2]tony@uib.ac.id, [3]2031016.jonathan@uib.edu

Abstract

Writing Mandarin characters is considered the most challenging component for beginners due to the rules and character formations. This paper explores the potential of a machine learning-based digital learning tool for writing Mandarin characters. It also conducts a comparative study between MobileNetV2 and MobileNetV3, exploring different configurations. The research follows the Multimedia Development Life Cycle (MDLC) method to create both the application and machine learning models. Participants from higher education institutions that offer Mandarin courses in Batam, Indonesia, were involved in a User Acceptance Test (UAT). Data was gathered through questionnaires and analyzed using the System Usability Scale (SUS) methods. The results show positive user acceptance, with an SUS score of 77.92%, indicating a high level of acceptability. MobileNetV3Small was also preferred for recognizing the user's handwriting, due to comparable accuracy size, rapid inference time, and smallest model size. While the application was well-received, several participants provided constructive feedback, suggesting potential improvements.

Keywords: *mandarin character writing; machine learning; digital learning tool; MobileNetV2; MobileNetV3; system usability scale*

1. Introduction

Language serves as a vital means of communication in human life, often ingrained through habitual use [1]. The diversity of languages from one country to another requires the establishment of common languages to facilitate idea exchange, organizational collaboration, and overseas administration [2]. Mandarin, an official language with global usage, holds particular prominence. Mandarin, which translates to "northern regional language" (北方語), is spoken by people in the northern and southwestern regions of China, reaching over 1 billion speakers [3], [4]. This widespread usage underscores its importance in global trade and the demand for Mandarin proficiency, particularly in fields such as economics, education, and technology.

Learning Mandarin involves four essential components: listening (听力 tīngli), speaking (口语 kǒuyǔ), reading (阅读 yuèdú) and writing (写作 xiězuò). Among these skills, writing Mandarin characters (漢字 hánzì) is considered the most challenging for beginners. This complexity arises from the nature of character formation and the rules of writing Mandarin characters [5], [6]. This challenge is not exclusive to novice learners; even individuals with some level of proficiency encounter it.

The shift from traditional handwriting to digital typing in daily life also contributes to a decline in handwriting literacy rates [7]. There are common instances where one may have a sense of familiarity with a character, but struggle to recall its specific components. This situation is commonly referred to as a "tip-of-the-pen" state or "character amnesia" [8]. Some research highlighted the potential solutions, such as adopting interactive methods [9] and incorporating mobile applications in class [10] to address these issues. Thus, the potential of new tools to support learners in this aspect can be examined.

In recent years, the development of artificial intelligence has also become a point of interest. According to a systematic review by [11], AI is recognized for its potential to bring significant benefits to human life, one of which is the education sector [12]. In its application, AI such as image recognition is used by several researchers to explore its potential in leveraging the quality of education and experimenting with various machine learning algorithms. One of

Accepted: 25-10-2023 | Received in revised: 21-12-2023 | Published: 26-12-2023

which was in the form of Handwritten Character Recognition (HCR) as a digital training tool to write foreign language characters [13], [14]. Hence, the potential for adopting image recognition technology as a digital training tool can be explored further.

Convolutional Neural Networks (CNNs) have been a widely adopted method in image recognition, especially for tasks like identifying handwritten characters. CNNs consist of convolutional layers, input neurons, hidden layers, and output layers, which work together to categorize input data into predefined classes [15], [16]. Several studies have highlighted the usage of CNNs in recognizing handwritten content, such as Bangla characters [17], [18], handwritten digits [19], [20], Arabic handwriting [21], [22], Mandarin handwriting [22], [23], and more. Instead of training a particular model from scratch, transfer learning (TF) approaches have been frequently employed by researchers, such as MobileNetV2 and MobileNetV3 to allow for rapid development and performance improvement [24], [25].

MobileNetV2 is a type of Convolutional Neural Network (CNN) architecture that is well-known for its lightweight and efficient design, which incorporates the concept of a linear bottleneck and an inverted residual structure to create more efficient layer designs [26]. This architecture consists of a 1x1 expansion convolution, followed by depth-wise convolutions, and finalized with a 1x1 projection layer. A residual connection is also established to connect the input and output only when they share the same number of channels. This design ensures the input and output can maintain a compact representation while increasing its capacity to make complex transformations for each channel individually in higher-dimensional space. Numerous research studies have demonstrated its promising results in terms of accuracy and model size when compared to other pre-trained models. This makes it recommended for devices with limited computational resources [25], [27].

As the next iteration of the MobileNets architecture, MobileNetV3 was recently introduced to cater to the requirements of both high and low-resource computing scenarios [26], [28]. To search for the best kernel size and adaptability for different hardware platforms, MobileNetV3 utilizes the NetAdapt algorithm, a new Network Architecture Search algorithm (NAS) [29]. This architecture also adopts a new nonlinearity version of a sigmoid function called Hard Swish (h-swish), which reduces model complexity and size, thereby reducing the number of training parameters, and contributing to overall model efficiency [26].

However, there have been mixed research findings when it comes to the performance of both MobileNetV2 and MobileNetV3. Some cases have shown MobileNetV3 to perform well [30], [31], while others have found it to be less accurate although within a small margin [30], [32]. Given the limited amount of research available on the comparison between MobileNetV2 and MobileNetV3 in the context of recognizing Mandarin character handwriting, there is an opportunity to explore this further.

The focus of this study is to evaluate the potential and effectiveness of a machine learning-based application as a digital writing learning tool for aiding Hanzi learners in higher education institutions, particularly in Batam, Indonesia. Additionally, the study will compare the performance of MobileNetV2 and MobileNetV3 architectures in recognizing a handwritten Mandarin character dataset.

2. Research Methods

The research method chosen for this study is the Multimedia Development Life Cycle (MDLC) method, similar to [33]–[36], as illustrated in Figure 1.

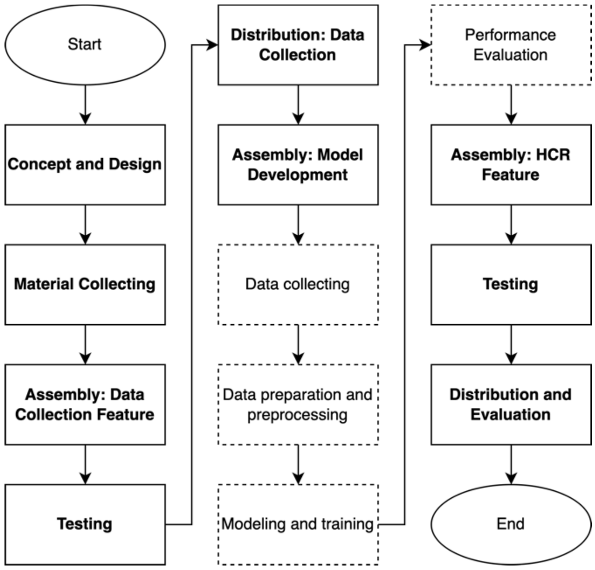

Figure 1. Research diagram

2.1 MDLC Stages

The concept phase encompasses a general overview of the application's purpose, such as main features, target users, platform, and more. This stage also includes the literature review from various sources to formulate research problems. The main concept of this application would be an application that consists of multiple levels and a drawing canvas to assess learners' handwriting corresponding to each predefined character. The application will also involve a machine learning model to support the application, similar to [34], [37], [38].

In the subsequent design phase, low-fidelity and high-fidelity prototypes will be created using Figma to provide a clearer picture of the application's appearance and functionality. The list of features and the flow of the application will also be determined during this stage. Based on the previous stage, the application will

DOI: https://doi.org/10.29207/resti.v7i6.5505

contain specifically the main menu scene and canvas scene that was designed with user-friendliness in mind.

the assets utilized in the material collecting process were both sourced from freely available online sources, as well as custom assets designed using Figma. These assets include sound effects, icons, and fonts. To achieve more favorable outcomes with our input image, we collected a dataset with the assistance of several volunteers, totaling up to 82 characters (50 images for each character) from public HSK1 vocabulary, refer to Table 1 for details and Figure 2 for examples. These were chosen because it is a monosyllabic character that fits into one writing canvas that contains a guideline. It's important to note that this phase operates asynchronously with the next phase as required.

Table 1. Summary of the collected Hanzi

Hanzi	Pinyin	Hanzi	Pinyin	Hanzi	Pinyin	Hanzi	Pinyin
一	yī	下	xià	书	shū	去	qù
二	èr	大	dà	了	le	叫	jiào
三	sān	小	xiǎo	些	xiē	号	hào
四	sì	人	rén	他	tā	吗	ma
五	wǔ	买	mǎi	会	huì	听	tīng
六	liù	来	lái	住	zhù	呢	ne
七	qī	吃	chī	你	nǐ	和	hé
八	bā	看	kàn	写	xiě	哪	nǎ
九	jiǔ	做	zuò	冷	lěng	喂	wèi
十	shí	不	bù	几	jǐ	喝	hē
回	huí	少	shǎo	本	běn	菜	cài
在	zài	岁	suì	水	shuǐ	说	shuō
坐	zuò	年	nián	点	diǎn	请	qǐng
块	kuài	开	kāi	热	rè	读	dú
多	duō	很	hěn	爱	ài	上	shàng
太	tài	想	xiǎng	狗	gǒu	这	zhè
她	tā	我	wǒ	猫	māo	那	nà / nèi
好	hǎo	是	shì	的	de	都	dōu
字	zì	月	yuè	能	néng	里	lǐ
家	jiā	有	yǒu	茶	chá	钱	qián
谁	shéi / shuí	个	gè				

Figure 2. Several examples of the dataset

2.1.4. Assembly

The application will be built using the Unity Game Engine and comprises three distinct stages: the data collecting feature, model development, and the Handwritten Character Recognition (HCR) feature.

The purpose of the data collection feature is to ease the data collection process. Volunteers' handwritten characters are collected from this feature and stored in a cloud storage bucket, subsequently used in the HCR model development.

The purpose of the model development is to create and assess the performance of MobileNetV2 and MobileNetV3 with various configurations. The selected model will be integrated into the subsequent assembly stage. Further details regarding this process are explained in section 2.2.

The purpose of the handwritten character recognition (HCR) feature is to evaluate the user's handwriting. To achieve this, a dedicated Python Flask server will be hosted on Google Cloud Run, aiming to minimize the overall size of the application. This server will utilize the selected machine learning model from the previous stage and execute the model to deliver prediction results based on the image sent to the server by the application.

The data collection feature and HCR feature from the assembly phase will undergo black box testing based on a set of test cases. The results will be assessed to determine whether the application aligns with the expected results of the application.

To ensure broad accessibility and installation of the application, the distribution will be made on two major mobile platforms: Android (Google Drive) and iOS (TestFlight) [38]. This decision aims to allow a wide range of volunteers and Hanzi learners within higher education institutions to access and assess this application. The usability of the final application will also be conducted towards participants detailed in sections 2.3 and 2.4.

2.2 HCR Model Development

2.2.1. Data Preparation and Preprocessing

As mentioned previously, the dataset will be collected with the help of the data collection feature in the application. Each image was sorted and filtered to remove any unfit handwriting. As a result, a total of 4100 images were labeled according to their corresponding characters and allocated into three distinct sets: 80% for training images, 10% for validation images, and the remaining 10% for testing images. Each image will belong to one of eighty-two classes. To facilitate the selected pooling layer discussed in point 2.2.2., the same dataset is inverted from black pixel to white pixel and vice versa, as shown in Figure 3.

The input values of the images were initially transformed from a scale of 0-255 to a scale of 0-1 to align with the input requirements of the MobileNetV2 and MobileNetV3 models. It is worth noting that both MobileNetV3Small and MobileNetV3Large have a default built-in preprocessing layer which will be

DOI: https://doi.org/10.29207/resti.v7i6.5505

deactivated in this process. Additionally, the training images were enhanced through the application of various Keras preprocessing methods, such as rotation, zooming, and shearing, as well as shifts in width and height. These augmentations provide the model with a wider range of diverse images to learn from. The weight of the individual classes was also uniform as they had an equal sum of images in the dataset.

Figure 3. Several examples of inverted characters

2.2.2. Modelling and Training

To ensure a fair comparison, both MobileNetV2 and MobileNetV3 architectures will utilize identical configurations for pooling and fully connected layers as indicated in Figure 4.

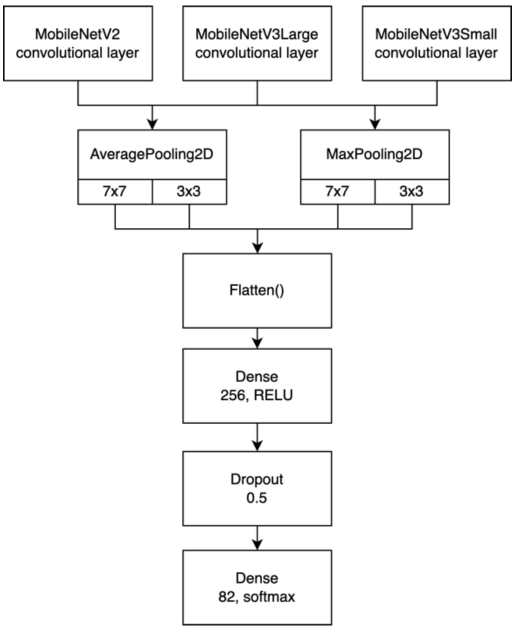

Figure 4. Model architecture

Average pooling operates by computing the mean value of elements (pixels) within the defined pooling region (kernel size) [39]. However, it tends to perform less effectively when a significant number of zero values are present in the pooling region, resulting in the inability to capture as many features as possible [40]. In contrast, max pooling evaluates the pooling region and selects the highest value within it [39]. For an image with a light background, the average pooling layer is better suited to capture more features than max-pooling [41]. Conversely, when dealing with an image featuring a dark background, a max pooling layer excels at capturing more features than average pooling [40]. With this, both pooling layers are considered with their respective datasets, and a combination with kernel sizes of both 3x3 and 7x7 will be explored.

The feature maps generated by the convolutional layers will be processed through a dense layer comprising 256 neurons with the Rectified Linear Unit (ReLU) activation function. Additionally, a dropout layer with a dropout rate of 0.5 will be applied to reduce the issues with overfitting by ignoring random neurons during the training process [42]. Finally, the last dense layer will use a softmax function to provide the classification outcome, consisting of 82 neurons representing 82 different classes.

These architectures were then compiled using categorical cross-entropy loss and the Adam optimizer. Afterward, the model will train using both training and validation datasets, incorporating several call-backs: early stopping, learning rate reduction (starting from 1e-4), and checkpoint, totaling up to 100 epochs.

2.2.3. Performance Evaluation

Following the training phase, the model will undergo testing using a separate batch of images that were not exposed to it during the training process. In the end, the model's performance will undergo an assessment, considering accuracy (refer to Formula 1), model size, and the times for training and inference. Notably, the testing will be done ten times to get the average inference time across all models. This evaluation will help determine its suitability for application.

$$Accuracy = \frac{Correct\ predictions}{Total\ number\ of\ predictions} \times 100\% \quad (1)$$

2.3 Participants

The study involved participants who were Hanzi learners enrolled in higher education institutions, specifically in Batam, Indonesia. Two universities were chosen due to the availability of Mandarin courses: Universitas Internasional Batam and Universitas Universal. The total population size from both universities, as per data from PDDIKTI in the second semester of 2022, amounted to 4,146 individuals. By applying the Slovin method for sample size determination, a sample size of a minimal 352 participants (excluding outliers) will be gathered.

2.4 User Acceptance Test (UAT)

To evaluate the potential and effectiveness of the application, the User Acceptance Test (UAT) will be conducted using a quantitative approach by distributing questionnaires to participants. Participants will be asked to answer a set of questions using a Likert scale ranging from 1 to 5, refer to Table 2 for details [43].

The result will be analyzed using the System Usability Scale (SUS) method based on the Formula 2 and concluded from Figure 5 [44].

DOI: https://doi.org/10.29207/resti.v7i6.5505

Table 2. Questionnaires statement

No.	Statements
1	I would like to use HanXie more frequently
2	HanXie is simple
3	HanXie is easy to use
4	HanXie could be used without the support of a technical person
5	Various functions in HanXie were well integrated
6	HanXie has a lot of consistency
7	Most people would learn HanXie very quickly
8	HanXie is very intuitive
9	I felt very confident in using HanXie
10	HanXie could be used without having to learn anything new

A validity test will be conducted on the respondents' answers using the Pearson Correlation formula. Following that, the reliability test will also be conducted using Cronbach's alpha. Both tests will be analyzed using the SPSS statistics software to determine whether the questionnaires are valid and reliable.

$$Y = \frac{P}{Q} \times 100\% \quad (2)$$

P is the sum of all respondent's scores, Q is the maximal possible sum of all respondents, and Y is the calculated SUS score.

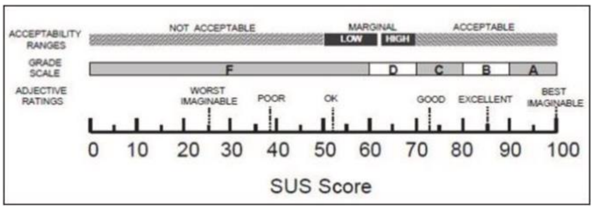

Figure 5. System Usability Scale (SUS)

3. Results and Discussions

3.1 Application Overview and Black Box Testing

Based on the user interface in Figure 6 and Figure 7, two main scenes were developed in Unity Game Engine using C# programming language and Visual Studio Community as the IDE. In the main menu scene, users are presented with an interface menu to select the characters that have been unlocked or previously finished. A level selector button was also implemented to ease the scrolling over a large content, specifically 99 different levels (82 individual characters and 17 tests).

Users will be presented with a canvas scene after clicking the practice button, as shown in Figure 7. By default, a guide text will only show on character levels to encourage learners to trace and practice their writing along the displayed characters. During test levels, the same canvas will be presented to learners, but without the guiding lines. The intention behind this is to challenge them to recall the characters they have practiced before advancing to the next level.

As illustrated in Figure 8, learners can choose a level from the menu. Once a character is written and the check button is clicked, the application will encode the hand drawing and send it to the server. Then, the server will process the image and return the prediction results. Subsequently, the application will display an action for the learner to follow.

Figure 6. Main menu scene

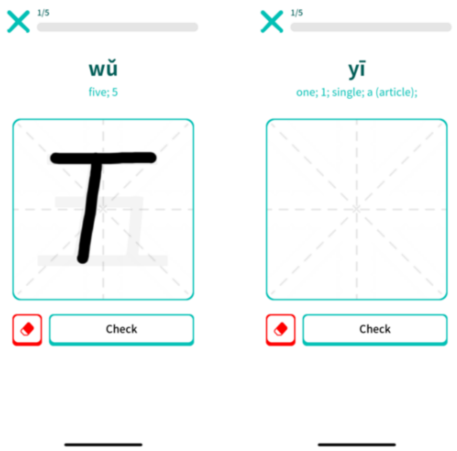

Figure 7. Canvas scene

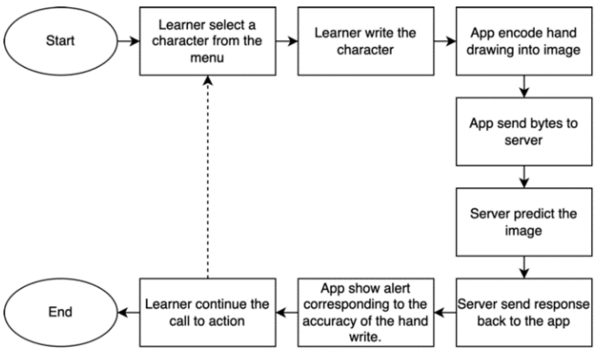

Figure 8. Application flowchart

A custom compute shader is used to portray the texture created by the user's hand drawing. The content was also managed using a series of scriptable object assets that were converted from a .xlsx file using a custom importer script. Finally, the black box testing method

DOI: https://doi.org/10.29207/resti.v7i6.5505

conducted towards the application shows satisfactory results as shown in Table 3.

Table 3. Black Box Testing

Test Case	Expected results	Results
Navigate the menu and start a level	Relevant levels will be set up and shown in the user interface.	Valid
Navigate the level selector and choose a level	Relevant levels will be shown in the user interface.	Valid
Save and reset progress	Progress will be saved or removed from the application.	Valid
Write specific character on the canvas and check the result	Results from the server will provide the next call to action through alerts (successfully or unsuccessfully).	Valid
Store an image to cloud storage	Task object will return a success response.	Valid
Send an image to Flask server	Response will be received with 200 code and prediction results.	Valid

3.2 HCR Model Evaluation

Table 4 provides a summary of different model configurations discussed in Section 2.2, focusing on MobileNetV2, MobileNetV3Large, and MobileNetV3Small, with varying pooling layers and kernel sizes. The results show that 3x3 kernels generally outperform 7x7 kernels, indicating that smaller kernels are better at capturing intricate patterns [45], like the handwriting in the datasets. Notably, some configurations don't reach 100 epochs due to the early stopping call-back function, which terminates training when no improvement is observed.

MobileNetV2 performs well in both configurations, achieving testing accuracies ranging from 99.76% to 100%. However, it has longer training times than the other two, similar to [26], averaging between 228s and 246s per epoch. MobileNetV3Large also performs well, with testing accuracy ranging from 96.59% to 99.76% with training times averaging around 194s to 239s. It's also interesting to note that using a 3x3 kernel size makes MobileNetV3Large smaller in size compared to MobileNetV2 despite having more parameters.

MobileNetV3Small, with fewer parameters, performs similarly to MobileNetV2 and MobileNetV3Large in terms of accuracy, especially when using 3x3 kernel size, achieving an accurate rate of 96.34%. Moreover, MobileNetV3Small has the fastest training and inference times with the smallest model size among the three.

Table 4. Models' performance overview

Model	Pooling	Kernel Size	Epoch	Parameters (M)	Test acc (%)	Train duration (h)	Colab inference duration (s)	Model size (MB)
MobileNetV2	Average	7x7	100	2.61	99.76	6.52	19.48	13.7
	Max	7x7	100	2.61	99.76	6.76	18.70	13.7
	Average	3x3	60	3.59	100	3.80	19.67	25.5
	Max	3x3	69	3.59	100	4.71	22.29	25.5
MobileNetV3Large	Average	7x7	100	3.26	96.59	6.44	20.41	16.0
	Max	7x7	100	3.26	97.32	6.65	17.96	16.0
	Average	3x3	100	4	99.76	5.85	21.34	24.9
	Max	3x3	85	4	99.76	4.58	19.1	24.9
MobileNetV3Small	Average	7x7	100	1.11	96.34	3.32	10.02	6.5
	Max	7x7	100	1.11	90.24	3.30	10.96	6.5
	Average	3x3	100	1.55	100	3.27	11.30	11.8
	Max	3x3	81	1.55	100	2.66	9.56	11.8

These results suggest that all the models perform exceptionally well and are great options for integration into the system, especially for mobile use cases. These outcomes also indicate that models trained with the dataset can maintain their effectiveness in unseen images (testing dataset). This is not surprising given that the dataset is specific to a predefined handwriting style. Selecting the appropriate model for the application requires consideration of various results. Therefore, when considering the results of all the models, MobileNetV3Small with a 7x7 kernel size and average pooling layer stands out as the preferred choice. This decision is based on its comparable testing accuracy, rapid inference times, and the advantage of having the smallest model size.

3.3 User Acceptance Evaluation

After the distribution phase, a set of questionnaires was distributed among the participants. Both a demo video and guidebook were also provided to ease the usage of the application for the participants. As 354 respondents answered the questionnaires, the validity and reliability tests were carried out. The analysis results revealed Cronbach's alpha value of 0.943, indicating the reliability of the results as it exceeds 0.6 [46]. Furthermore, the validity of the results is positive, as all the r_{count} are larger than r_{table} [47], as presented in Table 5.

Proceeding with the analysis of user acceptance, the SUS formula was used to calculate the acceptance of the model. Applying the total to the formula, the SUS score reached 77.92%, indicating a good and high level of acceptability. This shows that as a machine learning-

DOI: https://doi.org/10.29207/resti.v7i6.5505

based application, it is acceptable for the participants to support their writing learning process but with certain expectations towards them.

Table 5. Instrument results

	r_{count}	r_{table}	Cronbach's α	Conclusion	Sum
R1	.741	0.014	.943	Valid	1372
R2	.877	0.014		Valid	1394
R3	.868	0.014		Valid	1404
R4	.823	0.014		Valid	1382
R5	.766	0.014		Valid	1417
R6	.759	0.014		Valid	1420
R7	.820	0.014		Valid	1323
R8	.822	0.014		Valid	1397
R9	.855	0.014		Valid	1338
R10	.816	0.014		Valid	1345
Total					13792

This outcome reflects the feedback gathered from respondents about this application, which can be categorized into two main areas: features and performance. Some respondents expressed the need for more options related to HSK levels to enhance the application's utility. Others pointed out the application's potential for stroke order recognition and other display languages, such as Bahasa Indonesia. There were also suggestions for the addition of speaking and listening tests. It's worth noting that the initial container start-up time, which ranged from 7 to 10 seconds due to being hosted on a Google Cloud Run container, negatively impacted the user experience. Additionally, the application's size reached 83.6MB to support various mobile architectures, which is quite substantial for a simple application. Despite these constructive comments, there were also positive remarks about the application.

4. Conclusion

The study concludes with a positive indication of the machine learning-based application's potential and acceptance, as reflected by the System Usability Score (SUS) of 77.92% and constructive feedback among the participants.

The study also demonstrates that 3x3 kernels generally outperform 7x7 kernels across all models, emphasizing the effectiveness of smaller kernels in capturing intricate patterns, such as handwriting in the datasets. Despite MobileNetV2 has better testing accuracy, it comes at the cost of longer training times. MobileNetV3Large performs well with competitive accuracy and more efficient training times, while MobileNetV3Small, with fewer parameters, achieves a comparable accuracy of 96.34% while demonstrating the fastest training and inference times of 3.32 hours and 10.02 seconds respectively. It also achieves the smallest model size of 6.5MB.

Feedback from the participants and these insights can provide insights for researchers who explore the implementation of machine learning in the education sector, as well as offer guidance to educational institutions and application developers to explore better solutions.

References

[1] I. Markov, K. Kharitonova, and E. L. Grigorenko, "Language: Its Origin and Ongoing Evolution," *Journal of Intelligence 2023, Vol. 11, Page 61*, vol. 11, no. 4, p. 61, 2023, doi: 10.3390/JINTELLIGENCE11040061.

[2] R. Canestrino, P. Magliocca, and Y. Li, "The Impact of Language Diversity on Knowledge Sharing Within International University Research Teams: Evidence From TED Project," *Front Psychol*, vol. 13, 2022, doi: 10.3389/FPSYG.2022.879154.

[3] Y. Y. Su, R. Denadai, C. T. Ho, B. R. Lai, and L. J. Lo, "Measuring Patient-Reported Outcomes in Orthognathic Surgery: Linguistic and Psychometric Validation of the Mandarin Chinese Version of Face-Q Instrument," *Biomed J*, vol. 43, no. 1, pp. 62–73, 2020, doi: 10.1016/J.BJ.2019.05.011.

[4] V. Habic, A. Semenov, and E. L. Pasiliao, "Multitask Deep Learning for Native Language Identification," *Knowl Based Syst*, vol. 209, p. 106440, 2020, doi: 10.1016/J.KNOSYS.2020.106440.

[5] B. Lyu, C. Lai, C. H. Lin, and Y. Gong, "Comparison Studies of Typing and Handwriting in Chinese Language Learning: A Synthetic Review," *Int J Educ Res*, vol. 106, p. 101740, 2021, doi: 10.1016/J.IJER.2021.101740.

[6] Y. Yang *et al.*, "Brain Activation and Functional Connectivity during Chinese Writing: An fMRI Study," *J Neurolinguistics*, vol. 51, pp. 199–211, 2019, doi: 10.1016/J.JNEUROLING.2019.03.002.

[7] S. Huang, Y. Zhou, M. Du, R. Wang, and Z. G. Cai, "Character Amnesia in Chinese Handwriting: A Mega-Study Analysis," *Language Sciences*, vol. 85, p. 101383, 2021, doi: 10.1016/J.LANGSCI.2021.101383.

[8] S. Huang, W. Lin, M. Xu, R. Wang, and Z. G. Cai, "On the Tip of the Pen: Effects of Character-Level Lexical Variables and Handwriter-Level Individual Differences on Orthographic Retrieval Difficulties in Chinese Handwriting," *Quarterly Journal of Experimental Psychology*, vol. 74, no. 9, pp. 1497–1511, Mar. 2021, doi: 10.1177/17470218211004385.

[9] Y. Ying, D. Marchelline, and G. Wijaya, "Using Technology-Flashcard to Encourage Students Learning Mandarin," in *Journal of Physics: Conference Series*, IOP Publishing, Feb. 2021, p. 012138. doi: 10.1088/1742-6596/1764/1/012138.

[10] W. Zhou and X. Li, "Investigation of a Chinese Character Writing App: Learners' Perspectives," *Knowledge Management and E-Learning*, vol. 14, no. 1, pp. 15–29, Mar. 2022, doi: 10.34105/j.kmel.2022.14.002.

[11] A. Bozkurt, A. Karadeniz, D. Baneres, A. E. Guerrero-Roldán, and M. E. Rodríguez, "Artificial Intelligence and Reflections from Educational Landscape: A Review of AI Studies in Half a Century," *Sustainability 2021, Vol. 13, Page 800*, vol. 13, no. 2, p. 800, 2021, doi: 10.3390/SU13020800.

[12] M. Pikhart, "Intelligent Information Processing for Language Education: The Use of Artificial Intelligence in Language Learning Apps," *Procedia Comput Sci*, vol. 176, pp. 1412–1419, 2020, doi: 10.1016/J.PROCS.2020.09.151.

[13] M. Ponticorvo, E. Dell'Aquila, and R. Di Fuccio, "Hyper-Activity Books and Serious Games: How to Promote Experiential Learning beyond Distance," *International Journal of Environmental Research and Public Health 2022, Vol. 19, Page 11132*, vol. 19, no. 17, p. 11132, 2022, doi: 10.3390/IJERPH191711132.

[14] N. Bonneton-Botté *et al.*, "Can Tablet Apps Support the Learning of Handwriting? An Investigation of Learning Outcomes in Kindergarten Classroom," *Comput Educ*, vol. 151, p. 103831, 2020, doi: 10.1016/J.COMPEDU.2020.103831.

[15] M. Desai and M. Shah, "An Anatomization on Breast Cancer Detection and Diagnosis Employing Multi-layer Perceptron

DOI: https://doi.org/10.29207/resti.v7i6.5505

Neural Network (MLP) and Convolutional Neural Network (CNN)," *Clinical eHealth*, vol. 4, pp. 1–11, 2021, doi: 10.1016/J.CEH.2020.11.002.

[16] B. Kumeda, Z. Fengli, A. Oluwasanmi, F. Owusu, M. Assefa, and T. Amenu, "Vehicle Accident and Traffic Classification Using Deep Convolutional Neural Networks," in *2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing, ICCWAMTIP 2019*, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 323–328. doi: 10.1109/ICCWAMTIP47768.2019.9067530.

[17] R. R. Chowdhury, M. S. Hossain, R. Ul Islam, K. Andersson, and S. Hossain, "Bangla Handwritten Character Recognition using Convolutional Neural Network with Data Augmentation," in *2019 Joint 8th international conference on informatics, electronics & vision (ICIEV) and 2019 3rd international conference on imaging, vision & pattern recognition (icIVPR)*, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 318–323. doi: 10.1109/ICIEV.2019.8858545.

[18] S. N. Shuvo, F. Hasan, M. U. Ahmed, S. A. Hossain, and S. Abujar, "MathNET: Using CNN Bangla Handwritten Digit, Mathematical Symbols, and Trigonometric Function Recognition," in *Advances in Intelligent Systems and Computing*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 515–523. doi: 10.1007/978-981-15-7394-1_47.

[19] H. Kusetogullari, A. Yavariabdi, J. Hall, and N. Lavesson, "DIGITNET: A Deep Handwritten Digit Detection and Recognition Method Using a New Historical Handwritten Digit Dataset," *Big Data Research*, vol. 23, p. 100182, 2021, doi: 10.1016/J.BDR.2020.100182.

[20] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)," *Sensors*, vol. 20, no. 12, p. 3344, 2020, doi: 10.3390/S20123344.

[21] M. Elkhayati, Y. Elkettani, and M. Mourchid, "Segmentation of Handwritten Arabic Graphemes Using a Directed Convolutional Neural Network and Mathematical Morphology Operations," *Pattern Recognit*, vol. 122, p. 108288, 2022, doi: 10.1016/J.PATCOG.2021.108288.

[22] W. Liu, J. Wei, and Q. Meng, "Comparisons on KNN, SVM, BP and the CNN for Handwritten Digit Recognition," *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, pp. 587–590, 2020, doi: 10.1109/AEECA49918.2020.9213482.

[23] R. Ameri, A. Alameer, S. Ferdowsi, V. Abolghasemi, and K. Nazarpour, "Classification of Handwritten Chinese Numbers with Convolutional Neural Networks," in *Proceedings of the 5th International Conference on Pattern Recognition and Image Analysis, IPRIA 2021*, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/IPRIA53572.2021.9483557.

[24] H. M. Balaha *et al.*, "Recognizing Arabic Handwritten Characters using Deep Learning and Genetic Algorithms," *Multimed Tools Appl*, pp. 32473–32509, 2021, doi: 10.1007/S11042-021-11185-4.

[25] M. U. Hossain, M. A. Rahman, M. M. Islam, A. Akhter, M. A. Uddin, and B. K. Paul, "Automatic Driver Distraction Detection using Deep Convolutional Neural Networks," *Intelligent Systems with Applications*, vol. 14, p. 200075, 2022, doi: 10.1016/J.ISWA.2022.200075.

[26] A. Howard *et al.*, "Searching for MobileNetV3," in *Proceedings of the IEEE International Conference on Computer Vision*, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1314–1324. doi: 10.1109/ICCV.2019.00140.

[27] A. Nayak, S. Chakraborty, and D. K. Swain, "Application of Smartphone-Image Processing and Transfer Learning for Rice Disease and Nutrient Deficiency Detection," *Smart Agricultural Technology*, vol. 4, p. 100195, 2023, doi: 10.1016/J.ATECH.2023.100195.

[28] S. Qian, C. Ning, and Y. Hu, "MobileNetV3 for Image Classification," in *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 490–497. doi: 10.1109/ICBAIE52039.2021.9389905.

[29] M. Al-Sarem, M. Al-Asali, A. Y. Alqutaibi, and F. Saeed, "Enhanced Tooth Region Detection Using Pretrained Deep Learning Models," *International Journal of Environmental Research and Public Health 2022, Vol. 19, Page 15414*, vol. 19, no. 22, p. 15414, 2022, doi: 10.3390/IJERPH192215414.

[30] A. Hussain, B. Barua, A. Osman, R. Abozariba, and A. Taufiq Asyhari, "Performance of MobileNetV3 Transfer Learning on Handheld Device-based Real-Time Tree Species Identification," *2021 26th International Conference on Automation and Computing: System Intelligence through Automation and Computing, ICAC 2021*, 2021, doi: 10.23919/ICAC50006.2021.9594222.

[31] H. Tarek *et al.*, "Optimized Deep Learning Algorithms for Tomato Leaf Disease Detection with Hardware Deployment," *Electronics (Basel)*, vol. 11, no. 1, p. 140, 2022, doi: 10.3390/ELECTRONICS11010140.

[32] Y. Li, W. C. Huang, and P. H. Song, "A Face Image Classification Method of Autistic Children Based on the Two-Phase Transfer Learning," *Front Psychol*, vol. 14, 2023, doi: 10.3389/FPSYG.2023.1226470.

[33] R. Nurcholis, A. I. Purnamasari, A. R. Dikananda, O. Nurdiawan, and S. Anwar, "Game Edukasi Pengenalan Huruf Hiragana Untuk Meningkatkan Kemampuan Berbahasa Jepang," *Building of Informatics, Technology and Science (BITS)*, vol. 3, no. 3, pp. 338–345, 2021, doi: 10.47065/bits.v3i3.1091.

[34] V. D. Lestari and T. Huriah, "The Influence of Health Education Using Game-Based Learning Methods on Improving Smoking Prevention Behavior Among School-Age Children," *Jurnal Aisyah : Jurnal Ilmu Kesehatan*, vol. 7, no. 4, pp. 1267–1272, 2022, doi: 10.30604/jika.v7i4.1290.

[35] M. Ardiansyah and Riswanto, "Analisis Komparasi Ketertarikan Masyarakat Kota Batam Dalam Penggunaan Video Editor Capcut dan VN," *Jurnal Informasi dan Teknologi*, vol. 5, no. 3, pp. 91–102, 2023, doi: 10.60083/JIDT.V5I3.398.

[36] J. Pratama and R. I. Putri, "Design and Implementation of Animated Stickers as Educational Media Regarding COVID-19 Using the MDLC Method," *Science Tech: Jurnal Ilmu Pengetahuan dan Teknologi*, vol. 7, no. 2, pp. 1–12, 2021, doi: 10.30738/st.vol7.no2.a9190.

[37] M. N. I. Opu, M. R. Islam, M. A. Kabir, M. S. Hossain, and M. M. Islam, "Learn2Write: Augmented Reality and Machine Learning-Based Mobile App to Learn Writing," *Computers 2022, Vol. 11, Page 4*, vol. 11, no. 1, p. 4, 2021, doi: 10.3390/COMPUTERS11010004.

[38] R. Wibawa, A. Lokacarya, F. Kurniawan, and Y. Udjaja, "Japanese Language Learning Game 'Miryoku' using Android-Based Speech Recognizer API," *Procedia Comput Sci*, vol. 216, pp. 547–556, 2023, doi: 10.1016/J.PROCS.2022.12.168.

[39] P. Satti, N. Sharma, and B. Garg, "Min-Max Average Pooling Based Filter for Impulse Noise Removal," *IEEE Signal Process Lett*, vol. 27, pp. 1475–1479, 2020, doi: 10.1109/LSP.2020.3016868.

[40] N. Akhtar and U. Ragavendran, "Interpretation of Intelligence in CNN-Pooling Processes: A Methodological Survey," *Neural Comput Appl*, vol. 32, no. 3, pp. 879–898, 2020, doi: 10.1007/S00521-019-04296-5.

[41] Nurbaity Sabri, Haza Nuzly Abdull Hamed, Zaidah Ibrahim, and Kamalnizat Ibrahim, "A Comparison between Average and Max-Pooling in Convolutional Neural Network for Scoliosis Classification," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, pp. 689–696, 2020, doi: 10.30534/ijatcse/2020/9791.42020.

[42] Y. Gulzar, "Fruit Image Classification Model Based on MobileNetV2 with Deep Transfer Learning Technique," *Sustainability*, vol. 15, no. 3, p. 1906, 2023, doi: 10.3390/SU15031906.

[43] C. N. A. M. Gulfan and C. M. Vilela-Malabanan, "Evaluating the Usability and User Experience of Phytoplankton Cell

Counter Prototype," *Procedia Comput Sci*, vol. 197, pp. 309–316, 2022, doi: 10.1016/J.PROCS.2021.12.145.

[44] S. Bordoni and G. Tang, "Development and Assessment of a Contactless 3D Joystick Approach to Industrial Manipulator Gesture Control," *Int J Ind Ergon*, vol. 93, p. 103376, 2023, doi: 10.1016/J.ERGON.2022.103376.

[45] L. D. Tamang and B. W. Kim, "FVR-Net: Finger Vein Recognition with Convolutional Neural Network Using Hybrid Pooling," *Applied Sciences*, vol. 12, no. 15, p. 7538, 2022, doi: 10.3390/APP12157538.

[46] N. W. Raharjanti *et al.*, "Translation, Validity and Reliability of Decision Style Scale in Forensic Psychiatric Setting in Indonesia," *Heliyon*, vol. 8, no. 7, 2022, doi: 10.1016/j.heliyon.2022.e09810.

[47] S. R. Natasia, Y. T. Wiranti, and A. Parastika, "Acceptance Analysis of NUADU As E-learning Platform Using the Technology Acceptance Model (TAM) Approach," *Procedia Comput Sci*, vol. 197, pp. 512–520, 2022, doi: 10.1016/J.PROCS.2021.12.168.

DOI: https://doi.org/10.29207/resti.v7i6.5505