2020 Information Technology International Seminar (ITIS)
Surabaya, Indonesia, October 14-16, 2020

2020 6th Information Technology International Seminar (ITIS) | 978-1-7281-7726-7/20/$31.00 ©2020 IEEE | DOI: 10.1109/ITIS50118.2020.9321049

Performance Evaluation of Pre-trained Convolutional Neural Network for Milkfish Freshness Classification

Eko Prasetyo
Department of Informatics,
Engineering Faculty
University of Bhayangkara Surabaya
Surabaya, Indonesia
eko@ubhara.ac.id

Rani Purbaningtyas
Department of Informatics,
Engineering Faculty
University of Bhayangkara Surabaya
Surabaya, Indonesia
rpurbaningtyas@ubhara.ac.id

Raden Dimas Adityo
Department of Informatics,
Engineering Faculty
University of Bhayangkara Surabaya
Surabaya, Indonesia
dimas@ubhara.ac.id

***Abstract*— Milkfish are the top five fish of aquaculture products in Indonesia with high sales in traditional markets. Hence, the Indonesian people should recognize the freshness of the fish in the traditional market. An automated system to recognize the freshness of milkfish based on the eye using Convolutional Neural Network (CNN) deep learning requires vast image data in training sessions. For our small dataset, we performed transfer learning with fine-tuning pre-trained CNNs. In this study, we evaluate several pre-trained CNN models to classify milkfish eye freshness. The dataset consists of 234 milkfish eye images and three freshness class. The experiments and analysis results show that NasNet Mobile and Densenet 121 outperform state-of-the-art with the best performance on training, validation, and testing data.**

Keywords*— Convolutional Neural Network, milkfish eye, freshness, classification, transfer learning

I. Introduction

Indonesia is the highest fish producing country in Southeast Asia [1]. Data from the Ministry of Maritime Affairs and Fisheries in 2017 stated that the top five of Aquaculture Production by Major Commodities are Seaweed (69%), Nile Tilapia (7%), Shrimp (5%), Catfish (5%), and Milkfish (4%) from 16.11 million tons productions [2]. Simultaneously, milkfish is one of the product with high sales in traditional markets. Therefore, people should be able to recognize the freshness of the fish. Recognition of milkfish freshness for ordinary people is a difficult task without an automatic system.

Some automatic fish freshness recognition system has been developed, for example, [3] and [4] created an automatic system for detecting fish freshness based on the eyes and the gills, [5] used color image processing for detecting Common Carp fish freshness. Research in [6] used deep learning to recognize fish freshness. A non-destructive, simple, rapid, and low-cost fish freshness recognition tool should become more popular with the popularity of deep learning, as conducted by [6]. [7] used deep learning to detect and count the number of fish in the marine environment with complex image variations, including lighting, fish camouflage, background changes, shape deformations due to swimming fish. Therefore, this research is significant for classifying fish freshness. Here we focus on milkfish as the famous fish among Indonesian people.

Classification of fish freshness usually uses the image of body parts as a basis for classification, [3], and [8] using eyes and gills as a basis for recognition. In contrast, [4] and [9] uses only gills to classify fish freshness. The system developed by [3] uses the Red Green Blue (RGB), Hue Saturation Intensity (HSV), and L*a*b* color space as the main features while the artificial neural network (ANN) and support vector machine (SVM) as the classifier. The system achieves an accuracy performance of up to 97.33%. The color feature L*a*b* is also used by [8] to compare color difference features in the eyes and gills of fish with ice storage time. As a result, the system using the gills' color achieves better performance than the eyes, but the gills' recognition destroys the fish. Hence, non-destructive, simple, fast, and low-cost classification of fish freshness is appropriate when using eyes as a basis for recognition without touching or damaging fish. In [10], [11] uses only eyes to classify milkfish freshness with 18 color features of RGB, HSV, and L*a*b* color space, the system achieves accuracy to 69.94%. These studies use the image of the body part of the fish as the basis for the freshness classification but use a conventional method with specific steps. Therefore, we propose an innovation by examining the use of milkfish eye images to classify freshness using the Convolutional Neural Network (CNN). The images are being captured using a cellphone camera and then processed using CNN. We do not need other devices, such as transmitters and receivers, as additional wave features in the classification stage. Thus, our system is more straightforward, lighter, and faster.

From our explanation above, it is essential to report how our experimental results classify milkfish freshness using CNN as an integrated system between feature extraction and classification. In this study, we evaluated the Convolutional Neural Network (CNN) in classifying milkfish freshness based on the eyes. Hence, it would be a non-destructive, simple, fast, and low-cost automatic fish freshness classification system. However, to achieve high-performance, CNN requires vast image data for training the system classifying specific problems. For our small amount of image data, we use two strategies, data augmentation and transfer learning. Augmentation is a method for increasing data with various image modification variations: rotation, translation, scaling, shearing, zooming, and flipping.

We conduct transfer learning with fine-tuning from pre-trained CNN, for example, MobileNet [12], [13], ResNet [14], DenseNet [15], and NasNet [16]. The dataset used in this study is a milkfish eye dataset consisting of 234 images and three freshness classes (74 very fresh, 80 fresh, 80 not fresh). Milkfish samples are taken from traditional markets in Gresik City, East Java, Indonesia. Image acquisition is carried out on milkfish with ice storage time from the first day to the sixth day. We evaluate CNN performance through the performance of training, validation, and testing accuracy.

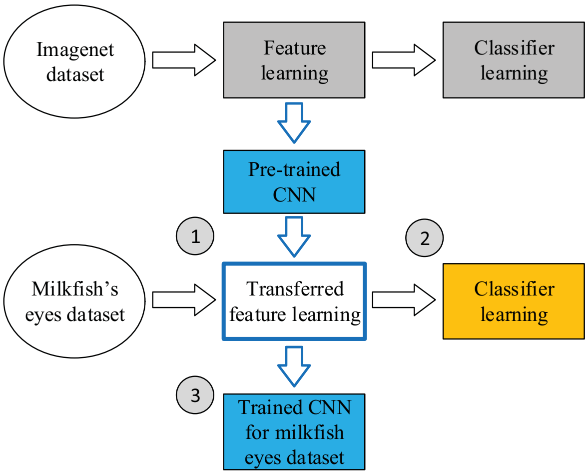


Fig. 1. Framework of transfer learning with milkfish eyes dataset

II. Research Methodology

A. Research Framework

We evaluate the performance of some CNNs by implementing transfer learning with fine-tuning, as presented in Fig. 1. We conducted a transfer learning from pre-trained CNN proposed by other research. Generally, pre-trained CNNs are trained using imagenet datasets consisting of millions of images and thousands of classes. The features and classifiers of pre-trained CNN are trained to recognize general classes, so we use these blocks features as knowledge transferred to our system. We use the features in the pre-trained CNN by conducting retraining some final layer.

The framework we use is as follows:

1. Transferred feature learning

   In this section, we investigated several pre-trained CNNs to be retrained with our dataset. We use only the feature blocks to solve our problems, while the classifier blocks are not used. Then, this block is retrained with fine-tuning to solve our problem.

2. Classifier learning

   We do not use the classifier of pre-trained CNN; instead of creating our classifier with two fully connected layers, as explained below.

3. Trained CNN for milkfish eye dataset

   As a result of the previous steps, we would gain CNN already trained on our dataset. Next, we compared the CNN in terms of training, validation, and testing sessions.

We propose a framework for classifying milkfish eye freshness using CNN as presented in Fig. 2, consisting of the following parts:

1. Feature learning

   This section is a block to train CNN features; when we apply transfer learning, we use weights from pre-trained CNN. Also, to apply fine-tuning, we retrain the last ten layers of each pre-trained CNN experimented while the rest are not retrained (frozen). We compare some pre-trained CNN including MobileNet V1, MobileNet V2, ResNet50, ResNet101, Densenet121, Densenet 169, Xception, and Nasnet Mobile.

2. Classifier learning

   In the classifier learning block, we use two fully connected layers, each consisting of 1024 neurons as hidden layers with ReLU activation and three neurons as the classifier with softmax activation.

3. Trained CNN

   The result of the training is a CNN model with weights according to milkfish eye classification. We evaluate the performance of all CNNs that have been retrained to meet an appropriate CNN architecture for solving our case.

B. Dataset

The milkfish eye dataset consists of 234 images and three freshness classes (74 very fresh, 80 fresh, 80 not fresh). Image size varies from 260 × 260 to 290 × 290 pixels with type jpg. Milkfish samples are taken from traditional markets in Gresik City, East Java, Indonesia. Image acquisition is carried out on milkfish with ice storage time from the first day to the sixth day. The image samples of the dataset are presented in Fig. 3. We divide the dataset with a proportion of 60:20:20 for training, validation, and testing, respectively.

C. Hyperparameters of CNN

We made hyperparameter adjustments during the experiment as follows: the number of neurons in the first and second fully connected layers was 1024 and 6, respectively, dropout 0.5, optimizer RMSprop, learning rate 1e-5, loss function categorical cross-entropy, epoch 300 times, batch size of data training and testing 10 and 6, respectively, and step-per-epoch 14.

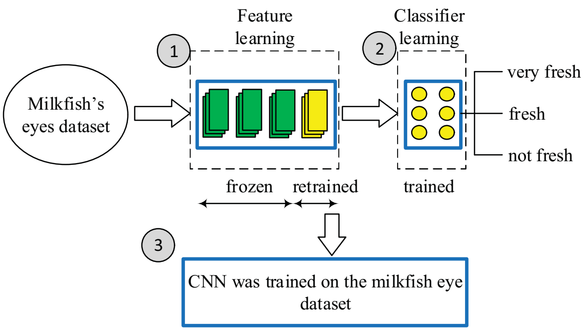
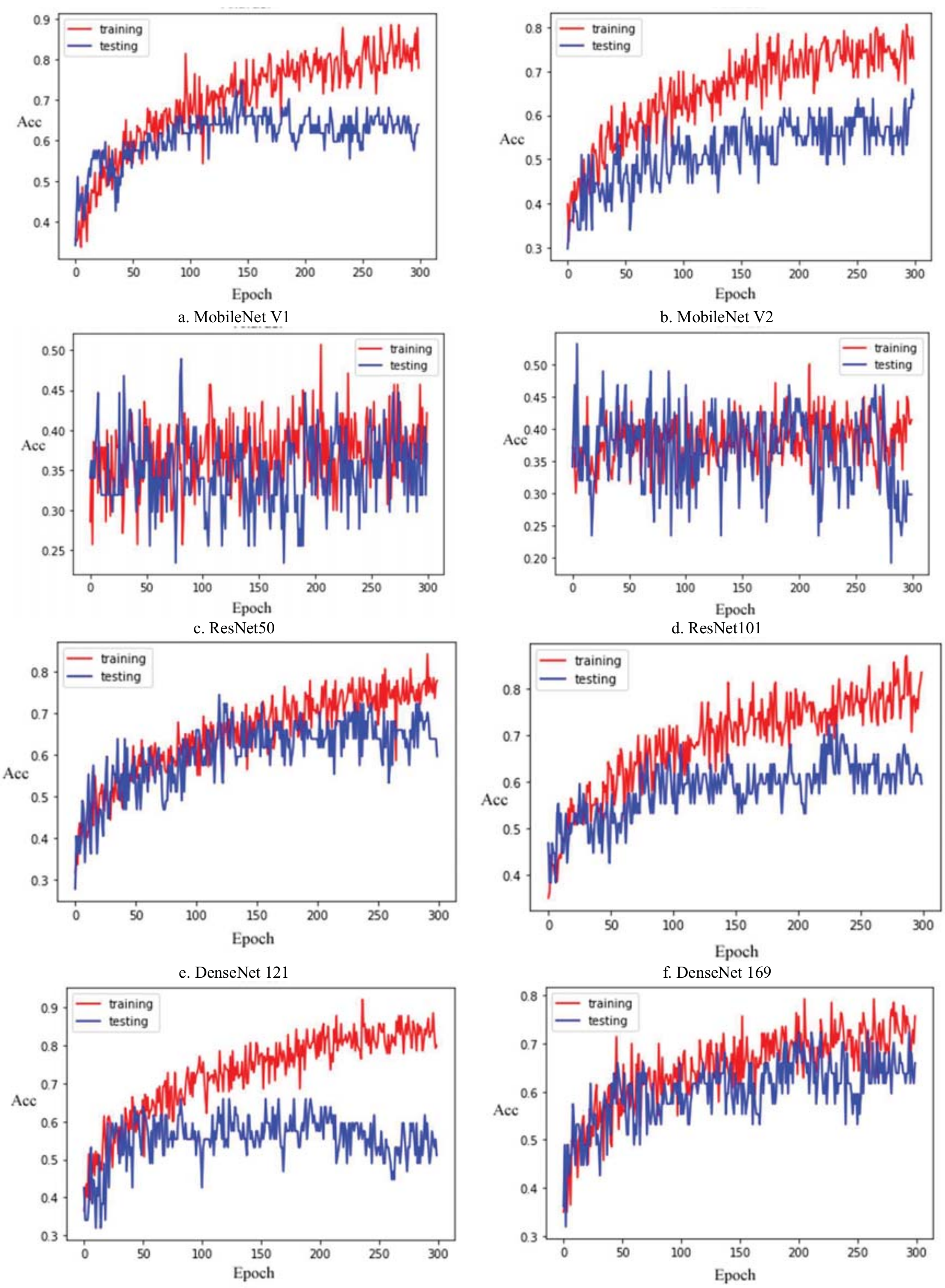


Fig. 2. Framework of milkfish eye freshness classification using CNN

a. Very fresh b. Fresh c. Not fresh

Fig. 3. Image samples of milkfish eye dataset



a. MobileNet V1
b. MobileNet V2
c. ResNet50
d. ResNet101
e. DenseNet 121
f. DenseNet 169
g. Xception
h. NasNet Mobile

Fig. 4. The performance of CNN during training

D. Data Augmentation

We faced a limited number of images; nevertheless, CNN needed large amounts of data. So we conduct image augmentation to add variation to the training data. We use augmentation as follows, rotation, shifting, shearing, zooming, and flipping.

TABLE I. THE BEST PERFORMANCE OF CNN AFTER 300 ITERATION

CNN	Training	Validation	Testing	Best Epoch
MobileNet V1	0.7286	**0.7447**	0.6382	146
MobileNet V2	**0.7786**	0.6596	0.5531	299
ResNet50	0.3571	0.4894	0.3617	82
ResNet101	0.3286	0.5319	0.5745	5
DenseNet 121	0.6714	**0.7447**	**0.6595**	120
DenseNet 169	0.7357	0.7234	0.6382	226
Xception	0.6857	0.6595	0.5532	83
NasNet Mobile	0.7571	0.7234	**0.6595**	201

III. RESULTS AND DISCUSSIONS

We experimented with several pre-trained CNNs to complete the milkfish eye classification, including MobileNet V1, MobileNet V2, ResNet50, ResNet101, Densenet 121, Densenet 169, Xception, and Nasnet Mobile. Using settings including hyperparameters, fine-tuning, training, validation, and testing data, as explained earlier, the training process obtained is presented in Fig. 4.

In general, there are three groups of training patterns as follows, overfitting, unrisen performance, and slowly rising performance. CNN with overfitting training are MobileNet V1 (Fig. 4 (a)), MobileNet V2 (Fig. 4 (b)), Densenet169 (Figu. 4 (f)), and Xception (Fig. 4 (g)). MobileNet experienced overfitting from epoch 130, where the accuracy of the training continued rising, but the validation accuracy could not rise again. MobileNet V2 has overfitting since epoch 40, where validation accuracy grows up, but validation is getting further away from training accuracy. Densenet 169 started overfitting since epoch 110, where the validation accuracy can no longer increase as training accuracy increases. Xception also has the same problem of overfitting since epoch 60. ResNet50 (Fig. 4 (c)) and ResNet101 (Fig. 4 (d)) have a bigger problem where CNN is not able to study the dataset; this can be observed from the performance of accuracy can not be increased. Even up to 300 iterations, the up and down accuracy ranges from 0.2-0.6. Whereas DenseNet 121 (Fig. 4 (e)) and NasNet Mobile (Fig. 4 (f)) provide different performance, accuracy continues to increase slowly to more than 0.7.

The results presented in Table 4 are the best performance achieved by all CNNs during 300 epochs. We recorded the best validation accuracy performance during the training session. The best epoch is an iteration where CNN achieves the best validation accuracy, while training accuracy is the accuracy of training data when validation achieves the best results. We use that model obtained for predicting testing data. Because testing data is data that CNN has never seen during a training session, we ensure that CNN is also robust at predicting new unseen data by examining the model using testing data.

Almost all CNNs try to improve performance during a training session; only ResNet50 is almost unable to improve performance because of overfitting occurrence. MobileNet V2 achieved the best validation performance at the final epoch of 299, but the validation accuracy of 0.6596 was still below Densenet 121 as the highest accuracy of 0.7447. Densenet 121 and MobileNet V1 achieved the best validation accuracy at epoch 120 and 146, respectively. In terms of training accuracy, MobileNet V2 achieves the best accuracy of 0.7786, while NasNet Mobile is 0.7571. The best model for classifying testing data is NasNet Mobile and DenseNet 121, where each achieves the same accuracy of 0.6595.

The performance shown by DenseNet 121 and NasNet is better than others because both have the same principle where there are two parallel convolutions, which then merge in the next convolution layer. This mechanism provides the advantage that lower-level features can be generated in parallel and brought together at the next layer. Without using this method, some of the lower level features are not all captured to affect the classification performance. Although ResNet also has a residual concept in the form of lower-level features added to the next layer, the residues sent are also lower-level features that have also been convoluted by the next layer. The features generated by Resnet are weak; therefore, they are outperformed by DenseNet and NasNet. The same problem also occurs in both MobileNet and Xception. In the future, we also need to modify or rearrange the CNN architecture with minimal parameters and work optimally on our and other datasets, for example, by incorporating the advantages of DenseNet with ResNet in order to improve optimal performance.

The experimental results show that DenseNet 121 and NasNet Mobile achieved the best performance among the state-of-the-art during the training stage. However, validation and testing accuracy cannot reach 0.8, at least have a performance improvement pattern along with training iteration. ResNet's results indicate that this architecture cannot study our dataset to classify milkfish's freshness.

IV. CONCLUSION

Our experiments show that NasNet Mobile and Densenet 121 outperforms other CNNs for milkfish's eyes freshness classification. During training, both have performed well on training and validation and achieved the best validation and testing with accuracy 0.7447 and 0.6595. Meanwhile, other CNNs with low performance require further investigation to find out the problem. Then we can determine the appropriate treatment applied to CNN improves performance. For example, by applying the strengths of the DenseNet concept to ResNet architecture in order to achieve better performance.

ACKNOWLEDGMENT

We would like to thank Directorate of Research and Community Service (DRPM) DIKTI that fund Authors' research in the scheme of Basic Research year 2020 at University of Bhayangkara Surabaya, with contract number 008/SP2H/AMD/LT/MULTI/L7/2020 on 10 June 2020, and 048/VI/AMD/LPPM/2020/UBHARA on 11 June 2020.

REFERENCES

[1] F. Firmansyah, S. Oktavilia, R. Prayogi, and R. Abdulah, "Indonesian fish consumption: an analysis of dynamic panel regression model," in International Conference on Tropical and Coastal Region Eco Development, 2019.

[2] C. for S. and I. Data, Marine and Fisheries in Figures 2018. 2018.

[3] H. Mohammadi Lalabadi, M. Sadeghi, and S. A. Mireei, "Fish freshness categorization from eyes and gills color features using multi-class artificial neural network and support vector machines," Aquacultural Engineering, vol. 90, p. 102076, Aug. 2020.

[4] A. Issac, M. K. Dutta, and B. Sarkar, "Computer vision based method for quality and freshness check for fish from segmented gills," Computers and Electronics in Agriculture, 2017.

[5] A. Taheri-Garavand Fatahi, S., Banan, A., Makino, Y., "Real-time nondestructive monitoring of Common Carp Fish freshness using robust vision-based intelligent modeling approaches," Computers and

Electronics in Agriculture, vol. 159, pp. 16–27, 2019.

[6] A. Taheri-Garavand, A. Nasiri, A. Banan, and Y. D. Zhang, “Smart deep learning-based approach for non-destructive freshness diagnosis of common carp fish,” Journal of Food Engineering, vol. 278, p. 109930, Aug. 2020.

[7] A. Jalal, A. Salman, A. Mian, M. Shortis, and F. Shafait, “Fish detection and species classification in underwater environments using deep learning with temporal information,” Ecological Informatics, vol. 57, p. 101088, 2020.

[8] M. Dowlati, S. S. Mohtasebi, M. Omid, S. H. Razavi, M. Jamzad, and M. De La Guardia, “Freshness assessment of gilthead sea bream (Sparus aurata) by machine vision based on gill and eye color changes,” Journal of Food Engineering, vol. 119, no. 2, pp. 277–287, Nov. 2013.

[9] M. K. Dutta, A. Issac, N. Minhas, and B. Sarkar, “Image processing based method to assess fish quality and freshness,” Journal of Food Engineering, 2016.

[10] E. Prasetyo, R. Purbaningtyas, and R. Dimas Adityo, “Cosine K-Nearest Neighbor in Milkfish Eye Classification,” International Journal of Intelligent Engineering and Systems, vol. 13, no. 3, 2020.

[11] E. Prasetyo, R. D. Adityo, and R. Purbaningtyas, “Classification of segmented milkfish eyes using cosine K-nearest neighbor,” in Proceedings of ICAITI 2019 - 2nd International Conference on Applied Information Technology and Innovation: Exploring the Future Technology of Applied Information Technology and Innovation, 2019, pp. 93–98.

[12] A. G. Howard et al., “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” Apr. 2017.

[13] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, “Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation Mark,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2018.

[14] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016.

[15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017.

[16] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning Transferable Architectures for Scalable Image Recognition,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2018.