ICSigSys

by Viny M

Submission date: 08-Nov-2020 03:10PM (UTC+0700)

Submission ID: 1439411375

File name: Zakhayu_Rian_Conference_ICSigSys_2019.pdf (1.48M)

Word count: 3622

Character count: 18829

2019 IEEE International Conference on Signals and Systems (ICSigSys)

Content-Based Image Retrieval using Convolutional Neural Networks

Zakhayu Rian, Viny Christanti, Janson Hendryli
Faculty of Information Technology
Tarumangara University
Jakarta, Indonesia
zakhayu.535150019@stu.untar.ac.id, viny@untar.ac.id, jansonh@fti.untar.ac.id

Abstract*—Searching a collection of images that have similarities with input images, without knowing the name of the image, makes a search system that applies the concept of content-based image retrieval (CBIR), is very necessary. In general, CBIR systems use visual features such as color, image edge, texture, and suitability of names in input images with images in the database. The method for classification is convolutional neural networks (CNN), while retrieval with cosine similarity. Dataset is divided into 5 masterclasses, each masterclass has 5 subclasses. The class used for retrieval is a masterclass, where the images of each large class are combined images of subclasses in the large class. From the experiments, we found that the CNN method has succeeded in supporting the retrieval task, by classifying image classes.

***Keywords*—cosine similarity, content-based image retrieval, convolutional neural networks, deep learning, VGG16**

1. Introduction

Images search has been very much done by technology companies. Examples of well-known companies are Google and Microsoft. As on Google, it provides an image search page on the Google Images page, while Microsoft has Bing Images. Both of them are tasked to be able to find a similar set of images based on their input in the form of images.

The process of searching images to display them, for example in both companies, is useful for displaying similar images. This function is very helpful for many users, in terms of looking for similar images based on the uploaded images, but is obtained from various other sources that have similar images, or images that have different viewpoints. This process is called content-based image retrieval (CBIR).

The method used to efficiently search for a collection of images in CBIR is to use digital images as inputs and use image class classification. Image classification is used to assist the retrieval process, by recognizing the type of image class so the machine will retrieve some digital images that match the input image class starting from the most similar ones based on the image class.

Some previous technique to classifying images, for content-based image retrieval is KNN (K-nearest neighbor algorithm), and the retrieval method is using color feature extraction [9].

In this paper, we are using a deep learning technique to support the CBIR classification. The deep learning method that we use is convolutional neural networks. The reason, why we use deep learning, is because it has representation learning [5].

Representation learning is a set of methods that allows a machine to be fed with raw data and to automatically discover the representations needed for detection or classification. Deep-learning methods are representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level. [11].

Deep learning convolutional neural networks have a nice reputation during the ImageNet competition. The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) has been running annually for five years (since 2010) and has become the standard benchmark for large-scale object recognition [8].

The experiment that has been tried on CBIR using CNN, by using a bag of word. This experiment has been done by Filip Radenović, Giorgos Tolias, Ondřej Chum from Czech Technical University in Prague. They experiment with 2 architecture, there are AlexNet and VGG. The achieved results are reaching the level of the best systems based on local features with spatial matching and query expansion while being faster and requiring less memory [4].

Another experiment that using CNN for classification is the classification for handwritten character recognition. This experiment has been done by Yann LeCun and his team. They are using CNN, to deal with the variability of 2D shapes. They are using MNIST [10].

The dataset used in the program is the iNaturalist, for the 2017 competition. This dataset used in the program will consist of 5 large classes, and each large class has 5 subclasses. In total there are 25 subclasses, which contain images of animals and plants.

The aim of this paper is to implement a convolutional neural networks method into a content-based image retrieval system. The retrieval tasks, are supported by classification, so only the images inside in the classified masterclass, the similarity will be calculated.

2. Convolutional Neural Networks

Convolutional networks have been tremendously successful in practical applications. The name "convolutional neural network" indicates that the network employs a mathematical operation called convolution. Convolution is a specialized kind of linear operation. Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers [5].

Convolutional neural networks or ConvNets is a neural network that uses a convolution method to extract activation values from a volume for another volume layer. ConvNets in a simple sense is a sequence of layers, where each layer of ConvNets, convert one activation volume to another volume,

978-1-7281-2177-2/19/$31.00 ©2019 IEEE

with different functions. There are 3 main types of layers that have three main layers, namely convolution layer (conv layer), pooling layer, and fully connected layer.

The forward pass stage consists of a convolution layer, in the convolution layer, an activation map is created, the result of the calculation of the dot product from the filter with the input volume. Next, activate the value with the ReLU function to reduce negative values, and pooling to reduce the size. This step is done many times and the number of repetitions of the process is determined freely.

The last step on the forward pass enters the fully connected layer where the output is made in vector form and is processed with its weight value. After obtaining the output from the fully connected layer, the softmax values and error values can be calculated for the values in the training dataset, to classify whether the output matches the model class or not.

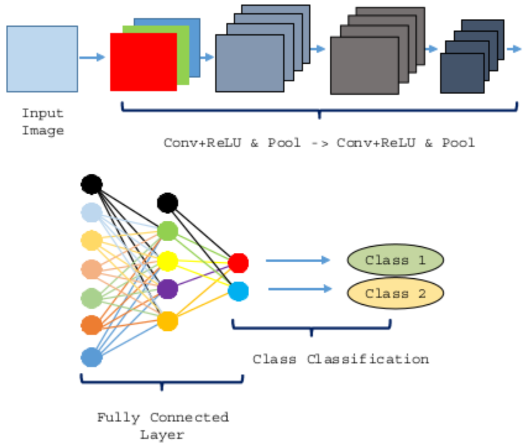


Fig. 1. Example of Forward Pass Scheme

Convolutional neural networks operate on image volumes. Thus, the input image can be said as the input volume. The volume consists of width (Width), height (Height), and dimensions of depth (Depth). The depth here is 3 colors, they are Red, Green, Blue (3 colors channel). On CNN, the input volume is initialized as W × H × D [6].

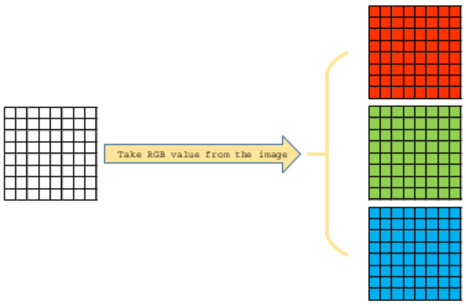


Fig. 2. Example Scheme for Taking the RGB Values from the Input Image

When convolution, the filter calculates the dot product, at each value in the input volume. The movement of the filter is to shift from the top to the bottom of the input volume, starting from the top left then to the top right. Every movement from left to right is done as much as stride. Stride is how many steps it convolutes.

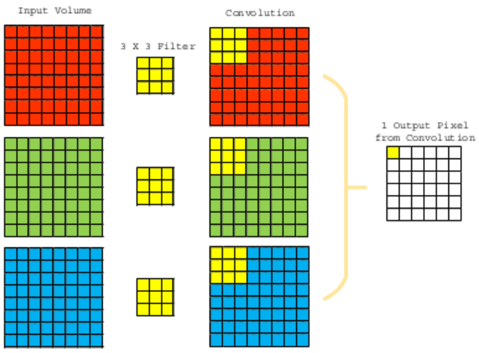


Fig. 3. Example Scheme for Dot Product Convolution

The rectified linear unit (ReLU) function is one of the activation functions on CNN. ReLU is a fast activation function because it only changes the negative pixel value to 0. The ReLU function is useful for reducing the computation of negative values, so the results of the training process only affect values greater than 0 [2].

Formula for ReLU:

$$\mathrm{ReLU}(x) = \max(0, x) \qquad (1)$$

Formula Information:

x is the input value.
If value $x \leq 0$ then $x = 0$
If value $x \geq 0$ then $x = x$

Next is pooling to reduce the size of the output volume to be smaller. The aim is to reduce the complexity of calculations in the program. The type of pooling commonly used is called max-pooling [6].

3. Cosine Similarity

When documents are represented as term vectors, the similarity of two documents corresponds to the correlation between the vectors. This is quantified as the cosine of the angle between vectors, that is, the so-called cosine similarity [1].

In this paper, it is not a document used in our cosine similarity, but the activation value of the model for each image. In cosine similarity, that is comparing the two vector distances to find the angle of difference, using the cos angle. To get a vector on each image that is searched for, in this paper we use the last fully connected layer (softmax layer).

4. VGG16 Model

The architecture for the model is the VGG16. VGG (Visual Geometry Group) model that has 16 layers of networks. The original VGG16 model created by Karen Simonyan and Andrew Zisserman, and the result in ILSVRC 2014 competition, the team secured the 2^{nd} place with a 7.3% error. The dataset they used, is the dataset from ILSVRC 2012. The model significantly outperforms the previous generation of models, which achieved the best results in the ILSVRC-2012 and ILSVRC-2013 competitions [7].

In this paper, the CBIR program using VGG16, and we do the training with datasets from iNaturalist, and our VGG16 which has been modified in the softmax layer section from 1000 to 25 classes due to amounts of classes in training.

5. Training the Testing Model Performance for Classification

The original iNaturalist dataset has a total of 13 masterclass, 5089 subclasses, and 579184 training images. But due to simplified the model performance evaluation, not all classes are included in the training. All unnecessary images in class like animal waste, footprint, animal carcass (destroyed carcass), blurred images, are deleted manually.

The chosen dataset consists of 5 masterclasses, and each of them has 5 subclasses, so the total class is 25 classes. Most of them are animals, but there is Plantae masterclass. The reason why we include Plantae class is to see how good our trained model performance, to classify between plants like images, and animal or insect class (most of the insect have more plants images in the background).

The proportion of dataset images are split into 2 datasets. The one for Training (70%), and the next one is for Validation (30%). Those images were split manually and randomly. The proportion of the dataset can be seen in **Table I**.

TABLE I. Proportion of Training and Validation Dataset

Masterclass	Subclasses	Training (70%)	Validation (30%)
Aves	*Agelaius phoeniceus*	1277	548
	Egretta thula	1256	538
	Fulica americana	1141	489
	Mimus polyglottos	1284	550
	Zenaida macroura	1228	526
Insecta	*Erythemis simplicicollis*	945	405
	Harmonia axyridis	918	393
	Junonia coenia	1041	446
	Pachydiplax longipennis	1079	463
	Vanessa atalanta	1008	432
Mammalia	*Canis latrans*	706	303
	Odocoileus hemionus	526	226
	Procyon lotor	545	233
	Sciurus carolinensis	1066	457
	Sciurus niger	1196	513
Reptilia	*Alligator mississippiensis*	416	179
	Anolis carolinensis	890	381
	Chelydra serpentia	640	274
	Crotalus atrox	583	250
	Trachemys scripta elegans	939	403
Plantae	*Achilliea millefolium*	832	357
	Eschscholzia california	689	295
	Fagus grandifolia	649	278
	Taraxacum officinale	756	324
	Toxicodendron radicans	617	265
Total Images		22227	9528

4.1 Model Performance

At this stage the training is applied, with 7000 epochs, the learning rate is 0,0001, Adam optimizer, and using categorical cross entropy for the loss. The total amount of time of training is 7:42:42.063126 (hh:mm:ss.ms).

The graphs of accuracy, the loss can be seen in **Fig. 4** and **5**. The classification report results displayed in **Table II**.

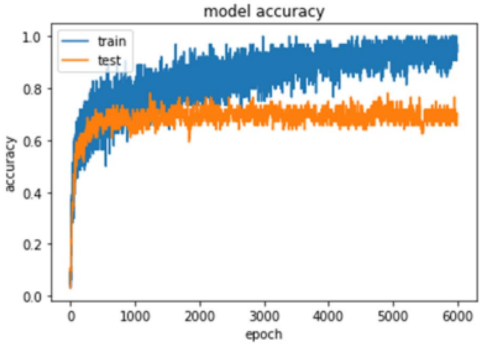


Fig. 4. Accuracy Graph for Model with 7000 Epoch, and learning rate 0.0001

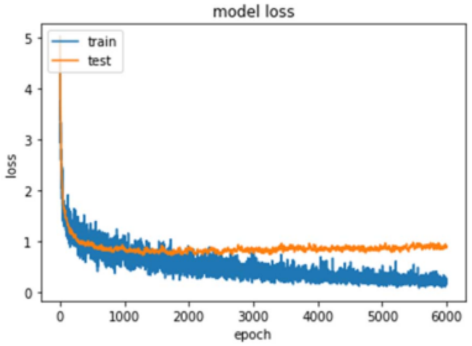


Fig. 5. Loss Graph for Model with 7000 Epoch, and learning rate 0.0001

From the graph, we can conclude that the accuracy in training, keeps increasing, close to the 1 value (100%), while the test which is the validation accuracy, stays between 0.6-0.7.

For the loss graph, we can conclude that the loss for the training, keeps decreasing as the accuracy keeps increasing. The test loss is the same too, it stays in the range between 1.3 – 1.4 until the last epoch.

The results of model classification report against the validation dataset are precision = 0.73, recall = 0.72, f1-score = 0.73.

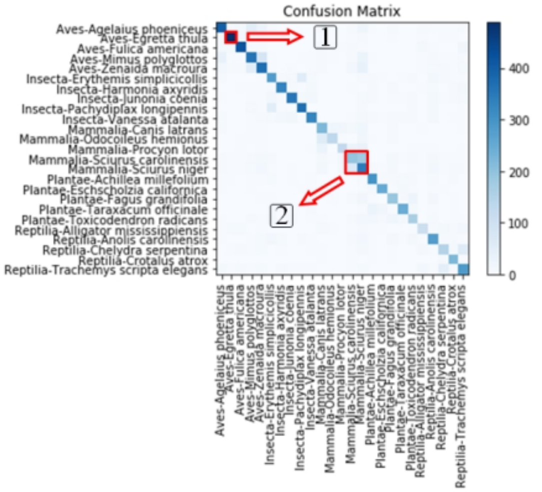


Fig. 6. Confusion Matrix for Model with 7000 Epoch, and learning rate 0.0001

From the confusion matrix, it can be seen that, for the most part, the images inside validation directory have been correctly predicted for the actual class. Most of Aves classes are correctly predicted. The most correct class is *Egretta thula* class marked by number "1". There are 2 classes that are still often confused with the results of their classification, they are *Sciurus carolinensis* and *Sciurus niger* class, marked by number "2".

6. Testing Model Performance for Image Retrieval

Testing the model for retrieval, the test images are from the internet (outside from iNaturalist dataset). Each subclass is tested twice with 2 different images. The retrieved images from the masterclass contain 250 images. The test results obtained by the average precision is 89.6%, and recall is 17.92% (maximum value for the recall is 20%).

Image relevance is seen from the suitability of the masterclass and subclasses category. If the classification of the masterclass is wrong and, if the classification of the masterclass is right but in the top 10 there is no image with the correct subclass, then the results of the precision and recall value for retrieval will be 0. The following are the results of image retrieval. We count precision and recall, for the top 10 only.

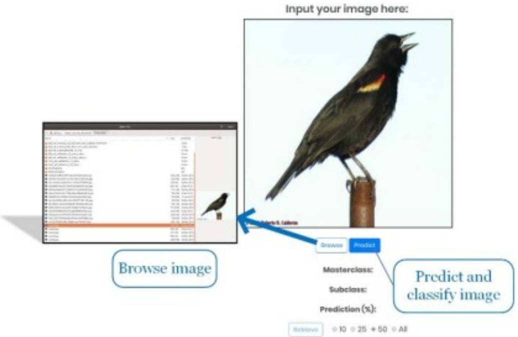
TABLE II. Proportion of Training and Validation Dataset

No	Input	*Precision* (%)	*Recall* (%)
1	*Agelaius phoeniceus* 1	100	20
2	*Agelaius phoeniceus* 1	100	20
3	*Egretta thula* 1	100	20
4	*Egretta thula* 2	100	20
5	*Fulica americana* 1	100	20
6	*Fulica americana* 2	100	20
7	*Mimus polyglottos* 1	100	20
8	*Mimus polyglottos* 2	100	20
9	*Zenaida macroura* 1	100	20
10	*Zenaida macroura* 2	100	20
11	*Erythemis simplicicollis* 1	100	20
12	*Erythemis simplicicollis* 2	100	20
13	*Harmonia axyridis* 1	100	20
14	*Harmonia axyridis* 2	100	20
15	*Junonia coenia* 1	100	20
16	*Junonia coenia* 2	100	20
17	*Pachydiplax longipennis* 1	100	20
18	*Pachydiplax longipennis* 2	100	20
19	*Vanessa atalanta* 1	100	20
20	*Vanessa atalanta* 2	100	20
21	*Canis latrans* 1	100	20
22	*Canis latrans* 2	100	20
23	*Odocoileus hemionus* 1	100	20
24	*Odocoileus hemionus* 2	100	20
25	*Procyon lotor* 1	100	20
26	*Procyon lotor* 2	100	20
27	*Sciurus carolinensis* 1	100	20
28	*Sciurus carolinensis* 2	0	0
29	*Sciurus niger* 1	100	20
30	*Sciurus niger* 2	100	20
31	*Achillea millefolium* 1	0	0
32	*Achillea millefolium* 2	100	20
33	*Eschscholzia californica* 1	100	20
34	*Eschscholzia californica* 2	100	20
35	*Fagus grandifolia* 1	100	20
36	*Fagus grandifolia* 2	0	0
37	*Taraxacum officinale* 1	100	20
38	*Taraxacum officinale* 2	0	0
39	*Toxicodendron radicans* 1	80	16
40	*Toxicodendron radicans* 2	100	20
41	*Alligator mississippiensis* 1	100	20
42	*Alligator mississippiensis* 2	100	20
43	*Anolis carolinensis* 1	100	20
44	*Anolis carolinensis* 2	100	20
45	*Chelydra serpentina* 1	100	20
46	*Chelydra serpentina* 2	100	20
47	*Crotalus atrox* 1	100	20
48	*Crotalus atrox* 2	100	20
49	*Trachemys scripta elegans* 1	100	20
50	*Trachemys scripta elegans* 2	0	0

Average Total	89.6	17.92

7. USER INTERFACE DURING IMAGE CLASSIFICATION AND RETRIEVAL

These are our program user interface during classification and retrieval.



Masterclass:

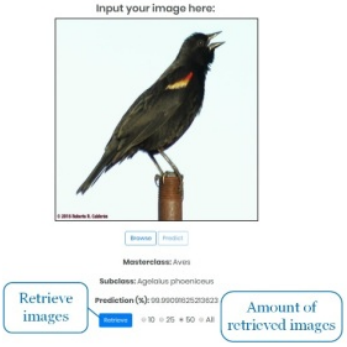
Subclass:

Prediction (%):

○ 10 ○ 25 ● 50 ○ All

Fig. 7. Browse Image and Start Classification

During the classification process, we browse some input images and predict it using our model.



Masterclass: Aves

Subclass: Agelaius phoeniceus

Prediction (%): 99.99099825213623

○ 10 ○ 25 ● 50 ○ All

Fig. 8. Classification Result and Start Image Retrieval

The classification results will show up. The name of the masterclass, subclass, and prediction rate for the image. Next, to start the retrieval, the user can select how many images want to be displayed.

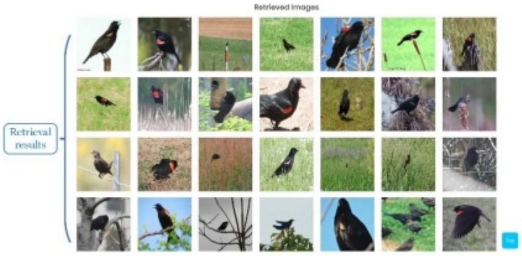
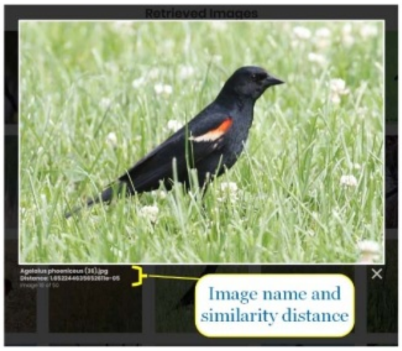


Fig. 9. Retrieval Result

After a few minutes, the retrieval will show all images as much as how many we choose the amount of image. All images are already sorted from the nearest similarity, until the farthest distances.



Agelaius phoeniceus (36).jpg

Distance: 1.6522446359552611e-05

Image 18 of 50

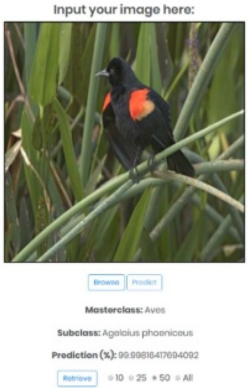
Fig. 10. Retrieved Image and Its Information

Each image, have 3 information there is the name of the image, distance value, and image position.

8. CLASSIFICATION AND RETRIEVAL EXAMPLE

Example of input image used during classification, and retrieval results.

Fig. 11. Classification results from input *Agelaius phoeniceus* 2 test image and correctly predicted with a 99% prediction rate.



Masterclass: Aves

Subclass: Agelaius phoeniceus

Prediction (%): 99.99816417694092

○ 10 ○ 25 ● 50 ○ All

Fig. 11. Input Image for *Agelaius phoeniceus* 2

Fig. 12. Retrieval results from input *Agelaius phoeniceus* 2 test image, and in the top 10 images, are correctly retrieved.

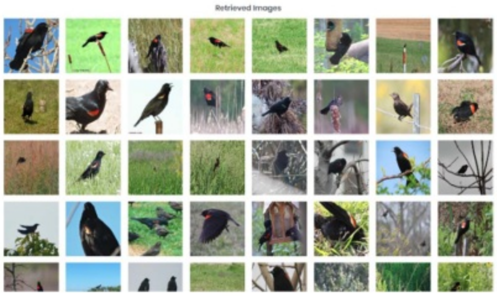
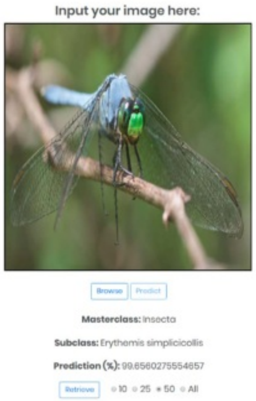


Fig. 12. Some Retrieved Images for *Agelaius phoeniceus* 2

Fig. 13. Classification results from input *Erythemis simplicicollis* 2 test image and correctly predicted with a 99% prediction rate.



Masterclass: Insecta

Subclass: Erythemis simplicicollis

Prediction (%): 99.6560275554657

○ 10 ○ 25 ● 50 ○ All

Fig. 13. Input Image for *Erythemis simplicicollis* 2

Fig. 14. Retrieval results from input *Erythemis simplicicollis* 2 test image, and in the top 10 images, are correctly retrieved.

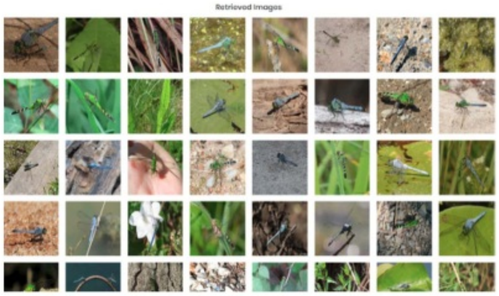
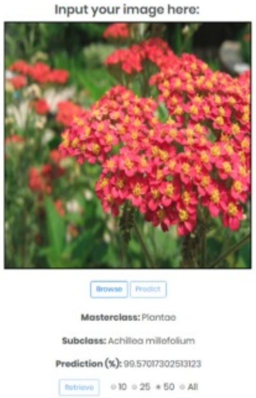


Fig. 14. Some Retrieved Images for *Erythemis simplicicollis* 2

Fig. 15. Classification results from input *Erythemis simplicicollis* 2 test image and correctly predicted with a 99% prediction rate.



Masterclass: Plantae

Subclass: Achillea millefolium

Prediction (%): 99.57017302513123

○ 10 ○ 25 ● 50 ○ All

Fig. 15. Input Image for *Achillea millefolium* 2

Fig. 16. Retrieval results from input *Erythemis simplicicollis* 2 test images, and in the top 10 images, are correctly retrieved. The color difference in the input image will still be considered relevant because it is also found in the training dataset, and in the same type of class.

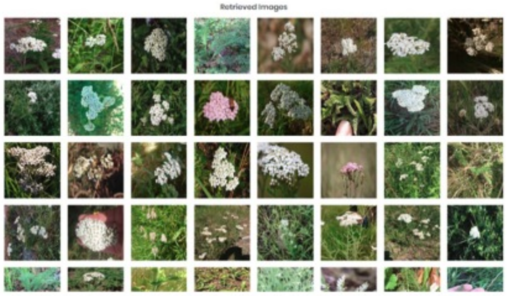
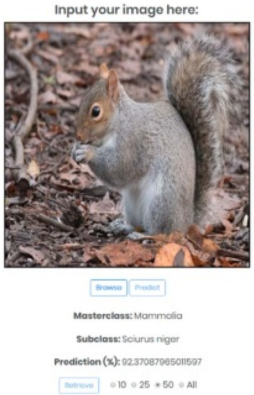


Fig. 16. Some Retrieved Images for *Achillea millefolium* 2

Fig. 17. The wrong example of classification when classifying *Sciurus carolinensis* 2, the model predicts it as *Sciurus niger* class.



Masterclass: Mammalia

Subclass: Sciurus niger

Prediction (%): 92.3708798501597

○ 10 ○ 25 ● 50 ○ All

Fig. 17. Input Image for *Sciurus carolinensis* 2

Fig. 18. Retrieval results from input *Sciurus carolinensis* 2 test images, and all top 10 images are not an inside correct subclass. So the precision and recall would be 0.

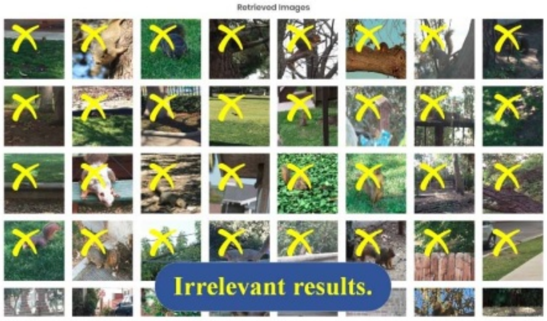
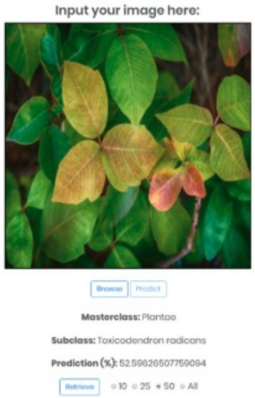


Fig. 18. Some Retrieved Images for *Sciurus carolinensis* 2

Fig. 19. Another wrong results is the *Toxicodendron radicans* 1. The subclass is correctly classified, but there are 2 irrelevant images in retrieval results, they are *Fagus grandifolia* class, marked “X” at **Fig. 20**.



Input your image here:

Masterclass: Plantae

Subclass: Toxicodendron radicans

Prediction (%): 52.59626507759094

○ 10 ○ 25 ● 50 ○ All

Fig. 19. Input Image for *Toxicodendron radicans* 1

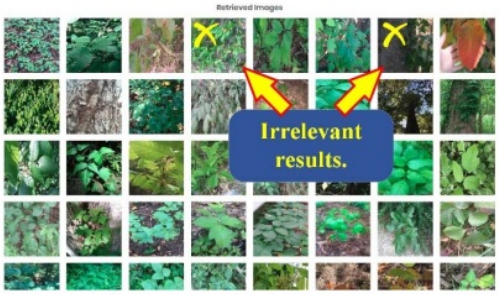


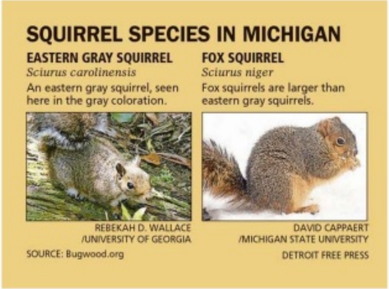
Fig. 20. Some Retrieved Images for *Toxicodendron radicans* 1 with 2 Irrelevant Images

9. Conclusion

Based on the experiment of model training, classification test, and image retrieval carried out on this CBIR program using the CNN method, it can be concluded that our trained VGG16 with 0.0001 learning rate, 7000 epochs, has succeeded in classifying the image in the validation dataset, with an accuracy (F1-score) of 73%, and an average of precision in retrieval is 89.6%.

At the stage of predicting the class or image classification in the validation dataset, there are two classes which are still often misclassified, namely the *Sciurus carolinensis* and *Sciurus niger classes*. This misclassified class can be seen in confusion matrix **Fig. 6**.

The two types of this class are squirrel animals, where the difference is only in color and size. For their colors, *Sciurus carolinensis* is dark gray, while *Sciurus niger* is yellow but slightly dark. For their sizes, *Sciurus niger* has a bit larger than *Sciurus carolinensis*. The size differences do not differ significantly, even seen with the human eye, because the squirrel is seen in the photo, where the size of the animal cannot be clearly seen how big the difference is.



An eastern gray squirrel, seen here in the gray coloration.

Fox squirrels are larger than eastern gray squirrels.

REBEKAH D. WALLACE /UNIVERSITY OF GEORGIA

DAVID CAPPAERT /MICHIGAN STATE UNIVERSITY

SOURCE: Bugwood.org

DETROIT FREE PRESS

Fig. 21. Comparison between *Sciurus carolinensis* and *Sciurus niger* Squirrel, Photographed by Detroit Free Press [3]

References

[1] A. Huang, "Similarity measures for text document clustering", Department of computer science, The University of Waikato, Hamilton, New Zealand, January 2008, p. 51.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", 2012, p. 3.

[3] Detroit Free Press, *Squirrel species in Michigan*. 2016 [Online]. Available: https://www.freep.com/story/news/local/michigan/2016/01/23/black-squirrels-in-michigan/78362460/. [Accesed: 2- Feb- 2019].

[4] F. Radenovic, G. Tolias, and O. Chum, "CNN image retrieval learns from BoW: unsupervised fine-tuning with hard examples", CMP, Faculty of Electrical Engineering, Czezch Technical University in Prague, ECCV16, 2016, p. 13.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, MIT Press, 2016.

[6] K. O'Shea, R. Nash, "An introduction to convolutional neural networks", Computer Vision and Pattern Recognition (CVPR), v. 2, December 2015, pp. 6-8.

[7] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", Computer Vision and Pattern Recognition (CVPR), Published as a conference paper at ICLR 2015, v. 6, April 2015, pp. 7-8.

[8] O. Russakovsky, J. Deng, H. Su, Jonathan Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, "ImageNet large scale visual recognition challenge", Computer Vision and Pattern Recognition (CVPR), v. 3, January 2015, p. 1.

[9] P. A. Deole, and R. Longadge, "Content based image retrieval using color feature extraction with KNN classification", IJCSMC, Vol. 3, Issue. 5, May 2014, p. 1274.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition", Proceedings of the IEEE, November 1998, pp. 1-10.

[11] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", Nature, Vol. 521, May 2015, pp. 436-444.

ICSigSys

ORIGINALITY REPORT

16%	11%	11%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijircce.com Internet Source	4%
2	www.nature.com Internet Source	2%
3	Submitted to University of Sunderland Student Paper	2%
4	docplayer.net Internet Source	1%
5	nzcsrsc08.canterbury.ac.nz Internet Source	1%
6	Submitted to Politeknik Negeri Bandung Student Paper	1%
7	ieeexplore.ieee.org Internet Source	1%
8	link.springer.com Internet Source	1%
9	Kevin Song Mardi, Viny Christanti Mawardi, Novario Jaya Perdana. "KNN Classification for	<1%

	CBIR with Color Moments, Connected Regions, Discrete Wavelet Transform", 2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE), 2019 Publication	
10	Teny Handhayani, Janson Hendryli, Lely Hiryanto. "Comparison of shallow and deep learning models for classification of Lasem batik patterns", 2017 1st International Conference on Informatics and Computational Sciences (ICICoS), 2017 Publication	<1%
11	ijarcsee.org Internet Source	<1%
12	Lecture Notes in Computer Science, 2016. Publication	<1%
13	www.ingentaconnect.com Internet Source	<1%
14	"Computer Vision – ECCV 2020", Springer Science and Business Media LLC, 2020 Publication	<1%
15	www.hindawi.com Internet Source	<1%
16	escholarship.org Internet Source	<1%

17	"Similarity Search and Applications", Springer Science and Business Media LLC, 2016 Publication	<1%
18	D. D. Pukale, S.G. Bhirud, V.D. Katkar. "Content-based Image Retrieval using Deep Convolution Neural Network", 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA), 2017 Publication	<1%
19	Filip Radenović, Giorgos Tolias, Ondřej Chum. "Chapter 1 CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples", Springer Science and Business Media LLC, 2016 Publication	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		