Image classification with Deep Convolutional Neural Network Using Tensorflow and Transfer of Learning

Aseel Sami Ali[1]
cs.19.56@grad.uotechnology.edu.iq

Matheel Emaduldin Abdulmunem[1]
110104@uotechnology.edu.iq

*[1] Department of Computer Science, University of Technology, Baghdad, Iraq

DOI: https://doi.org/10.36231/coedw/vol31no2.9

Received 2020/4/1 **Accepted** 2020/5/13

Abstract

The deep learning algorithm has recently achieved a lot of success, especially in the field of computer vision. This research aims to describe the classification method applied to the dataset of multiple types of images (Synthetic Aperture Radar (SAR) images and non-SAR images). In such a classification, transfer learning was used followed by fine-tuning methods. Besides, pre-trained architectures were used on the known image database ImageNet. The model VGG16 was indeed used as a feature extractor and a new classifier was trained based on extracted features.The input data mainly focused on the dataset consist of five classes including the SAR images class (houses) and the non-SAR images classes (Cats, Dogs, Horses, and Humans). The Convolutional Neural Network (CNN) has been chosen as a better option for the training process because it produces a high accuracy. The final accuracy has reached 91.18% in five different classes. The results are discussed in terms of the probability of accuracy for each class in the image classification in percentage. Cats class got 99.6 %, while houses class got 100 %.Other types of classes were with an average score of 90 % and above.

Keywords:
Convolutional Neural Network (CNN), Synthetic Aperture Radar (SAR), TensorFlow, Transfer learning, Visual Geometry Group (VGG16)

تصنيف الصور مع الشبكة العصبية التلافيفية العميقة باستخدام تينسورفلوونقل التعلم

أسيل سامي علي[1]
مثيل عمادالدين عبدالمنعم[1]
العراق بغداد، قسم علوم الحاسوب، الجامعة التكنولوجية،*[1]

الملخص

حققت خوارزمية التعلم العميق مؤخرًا الكثير من النجاح خاصة في مجال رؤية الكمبيوتر.يهدف البحث الحالي إلى وصف طريقة التصنيف المطبقة على مجموعة البيانات الخاصة بأنواع متعددة من الصور (صور الرادار ذي الفجوة المركبةSAR والصور ليست SAR) ، أستخدم نقل التعلم متبوعًا بأساليب الضبط الدقيق في مخطط التصنيف هذا . تم استخدام بنيات مدربة مسبقًا على قاعدة بيانات الصور المعروفهImageNet، تم استخدام نموذج VGG 16 بالفعل كمستخرج ميزات وتم تدريب مصنف جديد بناءً على الميزات المستخرجة .

تركز بيانات الإدخال بشكل أساسي على مجموعة البيانات التي تتكون من خمس فئات فئة صور الرادارSAR (المنازل) وفئات الصور ليستSAR (القطط والكلاب والخيول والبشر). تم اختيار الشبكة العصبية التلافيفية (CNN) كخيار أفضل لـعملية التدريب لانها نتجت عن دقة عالية. لقد وصلنا إلى الدقة النهائية بنسبة 91.18٪ في خمس فئات مختلفة. تتم مناقشة النتائج من حيث احتمالية الدقة لكل فئة في تصنيف الصورة بالنسبة المئوية. تحصل فئة القطط على 99.6٪ ، بينما تحصل فئة المنازل على 100٪ وتحصل انواع آخرى من الفئات بمتوسط درجات 90٪ وما فوق.

الكلمات المفتاحية:الشبكة العصبية التلافيفية (CNN)، الرادار ذي الفجوة المركبة (SAR) ،تينسورفلو ، تعلم النقل ،مجموعة الهندسة المرئية (VGG16).

1. INTRODUCTION

Recently, image classification is increasing and becoming a trend among technology developers especially with data growth in various parts of the industry such as e-commerce, automotive, healthcare and gaming. With just a few labeled images up to 98 percent accuracy, Facebook can now recognize your face with only a few marked images and label it into your Facebook folder. The technology itself nearly beats human ability in the classification or identification of images (Golob and Regan, 2001, pp.87-121).

Deep learning is one of the leading approaches to technology. Deep learning comes under the artificial intelligence umbrella, where it can behave or think like a human being. The device itself would usually be configured with hundreds of input data to make the training session more effective and swifter. It begins by giving a kind of' training' with all the input data (Patterson and Gibson, 2017).

Machine learning is also a commonly used system of image classification. Nevertheless, within machine learning, there are still pieces that can be changed. Therefore, the deep learning system will be occupied with the classification of images. When it comes to image recognition machine vision has its meaning. The combination of artificial intelligence software and computer vision technology will achieve the image classification outstanding result (Papernot, McDaniel, Jha, Fredrikson, Celik and Swami, 2016, pp.372-387).

The basic task of grading images is to ensure that all images are classified according to their particular sectors or groups (Li, VanRullen, Koch & Perona, 2002, pp.9596-9601).

Classification is simple for humans but computer problems have proved to be significant. Similar to detecting an object, it consists of unidentified patterns as it should be categorized into the proper categories. The numerous applications that use image recognition technology, such as vehicle navigation, robot navigation, and remote sensing. This appears to be difficult and there is a need for limited resources to develop this (Lai and Fox, 2010, pp.1019-1037).

Classification of images has been a significant obstacle in machine vision and thus has a long history. The challenge involves a wide variety of images multi-class due to color, size, environmental conditions and form. Big data of labeled training images are needed and to prepare this big data, it takes a lot of time and costs for training only (Jain, Nandakuma and Ross,2016, pp.80-105).

Transfer learning faced with the issue of achieving sufficient training data to reconstruct models. Transfer learning aims to transfer information from a large dataset to a smaller dataset called the target domain. If the functional spaces between the domain data are different, or the source and target tasks are focused on specific topics, enhancing target task output (Huang, Pan and Lei, 2017, p.907).

A Convolutional Neural Network (CNN) is a type of artificial neural network that is designed specifically to manipulate pixel data for image recognition and processing applications (Weimer, Scholz-Reiter and Shpitalni, 2016, pp.417-420).

CNN is efficient image processing, Artificial Intelligence (AI) that uses deep learning to execute both generative and descriptive tasks, often using image and video recognition machine vision(Hosny, Parmar, Quackenbush, Schwartz and Aerts, 2018, pp.500-510).

CNN uses a method that represents a multilayer perceptron designed to meet reduced processing needs. A CNN's layers consist of an input layer, an output layer and a hidden layer that contains several convolutional layers, pooling layers, fully connected layers, and layers of normalization. Removing limitations and increasing image processing capacity results in a method that is much more efficient, simpler to trains that are restricted for image processing (Acharya, Oh, Hagiwara, Tan, Adam, Gertych and San Tan, 2017, pp.389-396).

This paper will use a mixture of deep convolutional neural networks and transfer learning to achieve high-resolution image classification within a rapid period of SAR images and non-SAR images.

2. RELATED WORK

A brief overview of previous studies on image classification, using CNN and transfer learning for recognizing the SAR target.

2.1. Image classification:

Kamavisdar, Saluja and Agrawa (2013) explain using the Decision Tree (DT) as the image classification techniques. The DT has several datasets under each hierarchical classification. The calculation of membership for each of the groups must be completed. On the intermediate stages, the classifier allowed some rejection of the class. This method also required three parts, the first of which was to find terminal nodes and the second in class placement within it. Third, the nodes are partitioned. This approach is called an efficiency rate which is very simple and high (pp.1005-1009).

Pasolli, Melgani, Tuia, Pacifici and Emery (2013) clarify article addresses active learning on Support Vector Machine (SVM), which has been very strongly growing interest during that period. It also suggested some new ideas by integrating spatial with spectral information from a sequential phase in the trial process. It needs three methods, where the first is the distance from Euclidean. It measured some of the samples from the key spatial part of the study. The second technique is based on the technique of the Parzen window and eventually involves spatial entropy. The result indicated that two of the images are extremely resolute in terms of regular effectiveness (pp.2217-2233).

Gregor, Danihelka, Graves, Rezende and Wierstra (2015) show that Neural Network Architecture (NNA) as a method for grading images. The system consists of a mixture of mimics of two human-eye pairs and auto-encoding sequence variation. It involved a lot of complex images but the device gradually improves the Modified National Institute of Standards and Technology database (MNIST) models in the course of this analysis. The MNIST database is the software open-source to use as the training set. It also experiments with Street View House Numbers dataset where the outcome has been enhanced as it can not be identified even by the human eyes (pp.1502-04623).

Rastegari, Ordonez, Redmon and Farhadi (2016) illustrate an image classification system based on the structure of a CNN. The training was performed such that a balanced number of face images and non-face images were used for training by deriving additional face images from the face images data. The face classification method utilizes the CNN with 120 trained data and the auto-stage training produces a detection rate of 81.6% with just six false positives on the Face Detection Data Set and Benchmark (FDDB), where the existing state of the art hits a detection rate of about 80 % with 50 false positive (pp.525-542).

Korytkowski, Rutkowski and Scherer (2016) clarify a simple classification of images by boosting the Fuzzy Classifiers based on the journal. An easy way of distinguishing between known groups and unknown ones. This method simply boosts meta knowledge where the majority of local characteristics can be found. It was checked with some big image data and compared to the image model of the bag-of-features. The result offered much greater classification accuracy because it was a research phase that culminated in a short period where it generated 30% shorter than the previous one (pp.175-182).

Ahmed and Mahmoud (2018) illustrate using a previously trained model of convolutional neural networks based on certain parameters and hundreds of images to train and predict gender images using probability measurements. Accuracy of precision equals 0.68 and 0.3225, respectively (pp.1717-1732).

Abbas, R. F. (2020) shows that a new method inspired by the process of setting edges, one of the basic methods used in many fields, including radar images, which helps to view objects such as mobile vehicles, ships, aircraft, and meteorological and terrain types. To accurately recognize these objects it is important to detect their edges. The proposed methods (Ridgelet transform, Bezier curve and Sobel operator) with Ridgelet transform has been used in her research and has shown better results than wavelet transform; it eliminates noise by using ridgelet transform soft thresholding prior to edge detection. The findings show that the preferred approach has superior effects over Sobel edge detection and the wavelet process in both subjective and objective studies. Whereas in the proposed approach the Peak Signal to Noise Ratio (PSNR) values were increased to 12.6542, 12.9514, 12.8574 and 12.3013respectively (pp.185-192).

2.2. SAR Target Recognition with CNN's Learning :

Chen and Wang (2014) illustrate a single convolutional layer could successfully extract the feature specification of SAR targets using randomly sampled SAR target patches with unsupervised learning and reached 84.7% accuracy in 10-class rating tasks (pp.541–547).

Morgan (2015) explains using the architecture of three convolutional layers, using a fully connected Softmax layer as a classifier, improving the accuracy to 92.3% (p.94750).

Chen, Wang, Xu and Jin (2016) illustrate using an all-Convolutional network of five layers. In a convolutional layer, the authors followed a drop-out approach and removed the fully connected layer to prevent overfitting, as the minimal training data were inadequate to train CNN. The experiment results showed the state—efficiency of SAR target recognition in the MSTAR dataset, reaching an accuracy of 99.13% (pp.4806-4817) .

2.3. Transfer learning using CNN :

Oquab, Bottou, Laptev and Sivic (2014) explain the PASCAL VOC dataset proves the layers trained on ImageNet can be reused to effectively eliminate the mid-level image features (pp.1717–1724).

Shin, Roth, Gao, Lu, Xu, Nogues, Yao, Mollura and Summers (2016) demonstrate that the area of handling medical images where the data-poor also occurs. Transfer learning is an efficient method for using CNN to identify medical images with the help of appropriately annotated natural images. This paper by fine-tuning CNN models pre-trained from the natural image dataset accomplished two unique computer-aided detection problems in medical images. They further discussed various common CNN architectures and dataset sizes, concluding that there should be careful consideration of the trade-off between better learning models and the use of more training data. The classical pattern was common to lung tissue (pp.1285-1298).

Christodoulidis, Anthimopoulos, Ebner, Christe and Mougiakakou(2016) show that using pre-trained the network on six general texture databasesand fine-tuned on the target database after transferring particular layer numbers, reaching 2% performance improvement over the same network trained on the targets (pp.76-84).

3. THE PROPOSEDMETHOD

Based on Figure (1), it is the framework of image classification where deep convolutional neural networks are also applied. For all of this cycle, there are four stages. Each of the phases is included on TensorFlow as the open-source software and Python as its programming language. Then, the images of classes will be collected as (inputs). The process is then continued by downloading pre-trained Model Visual Geometry Group (VGG16) as a feature extractor and by applying CNN to train a new classifier based on extracted features, and eventually, all images will be categorized into their classes.

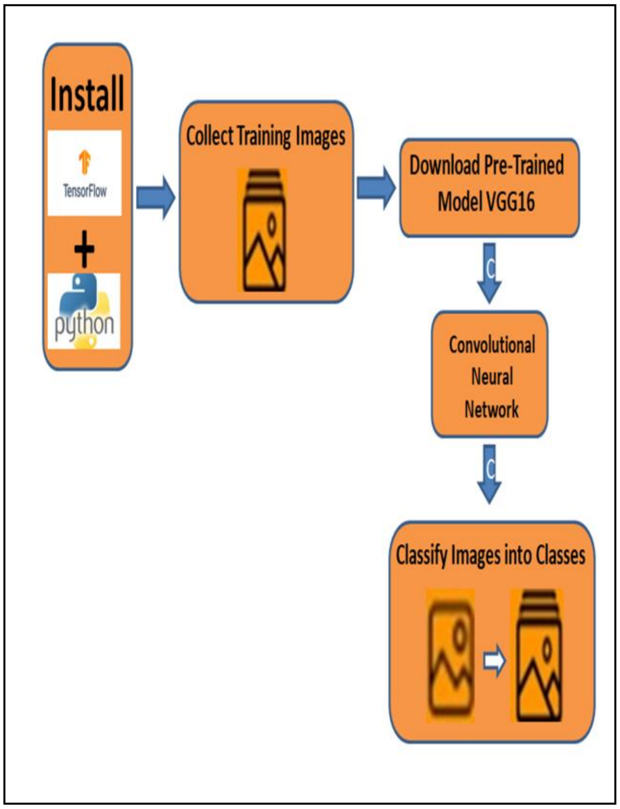

Install
TensorFlow
+
python
Collect Training Images
Download Pre-Trained Model VGG16
Convolutional Neural Network
Classify Images into Classes

Figure (1). The block diagram of Image classification.

4. TRAINING IMAGES

The total number of images as shown in Table (1) is (908).

Table (1).Number of images according to the type of classes.

No.	Type of Classes	Name of Classes	No. of Images
1.	Non SAR images	Cats	202
2.	Non SAR images	Dogs	202
3. .	Non SAR images	Hourses	202
4.	Non SAR images	Humans	202
5.	SAR images	Houses	100
Total			**908**

5. IMAGE CLASSIFICATION SYSTEM STEPS

A flowchart of the image classification system steps in Figure (2). The flowchart shows that the systems will be started by insert collecting images of the classes as input. These classes consist of five types which (cats, dogs, horses, humans, and houses) and each class will take label (0, 1, 2, 3 and 4) respectively. By specifying labels the learning becomes supervised that will help to clarify the classification is correct or not. After that, the Pre-trained model is downloaded to transfer learning and it will work as an extractor features. In the next step, CNN is applied as a trainer by training the model on the features extracted from the previous step then will generate a trained model that running for validation or testing. After that, the process ends after the output is classified into the right type of classes where CNN will work as a classifier for the tested image into the predicted class and if it is not the image of a particular class that supposedly acts as output then it needs to start over again from CNN. To test another image it needs to start again from CNN.

The result obtained from classifier in this flowchart are five values of probability for labels (0, 1, 2, 3 and 4) and the label of the highest probability is considered the predicted class for the tested image and by matching the specified label for the image in the training step with the label predicted from the classification process will decide the correctness of prediction or not.

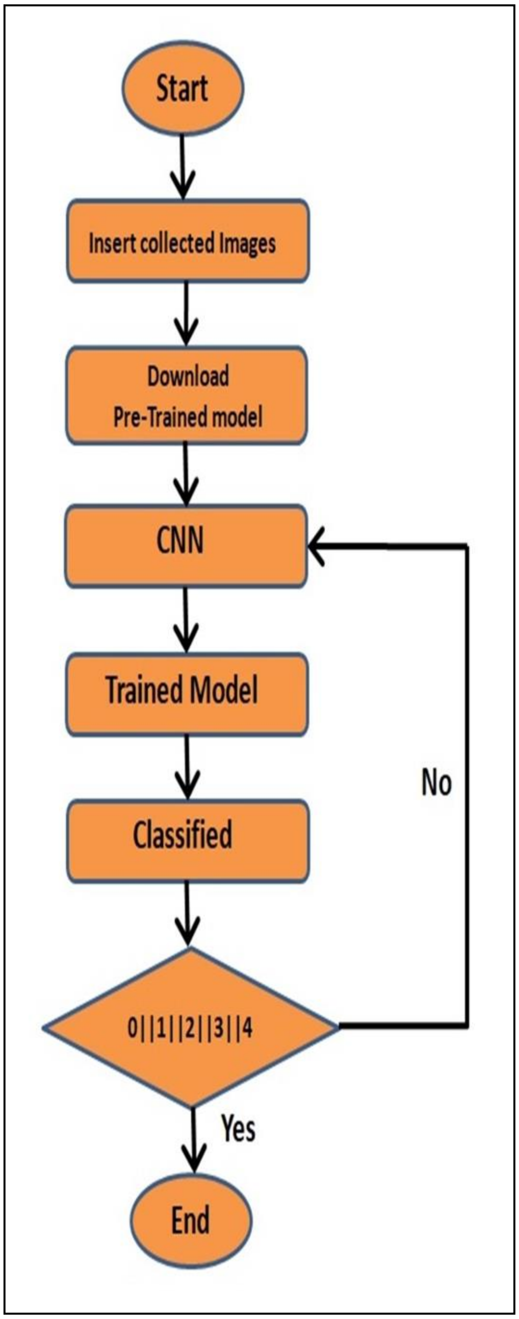


Figure (2)image classification system steps.

6. PERFORMANCE MEASURES

Performance measure on multiclass of image classification system can be evaluated by(Accuracy, F1 Score, Precision and Recall):

a) The Accuracy ratio uses to verify the experiment, it is calculated as the percentage of the total number of corrected predictions (both true positive and true negative) divided by the total number of samples examined as shown in Equation (1) (Baratloo, Hosseini,Negida, and El Ashal, 2015, pp. 48-49).

$$\text{Accuracy} = \frac{\mathrm{TP+TN}}{\mathrm{TotalNumbersofsamples}} \times 100 \qquad (1)$$

b) F1 score is metric for measuring the performance of machine learning model and good metric when data is imbalanced and the F1Score computed by as shown in Equation (2):

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (2)$$

c) Precision computed by the following equation:

$$\text{Precision} = \frac{TP}{TP+FP} \qquad (3)$$

d) Recall computed by the following equation:

$$\text{Recall} = \frac{TP}{TP+FN} \qquad (4)$$

Whereas:

- TP is the number of the images correctly identified prediction for each class.
- TN is the number of the images correctly rejected prediction for certain class.
- FP is the number of the images incorrectly identified predictions for certain class.
- FN is the number of the images incorrectly rejected predictions for certain class.

7.IMPLEMENTATION

7.1 .Loading and Preprocessing the Dataset.

As shown in Figure (3) built our dataset consisting of five classes, the first four classes (cats, dogs, horses and humans) representing non-SAR images downloaded from the Kaggle site, the last category is houses representing SAR images downloaded from the site (earthexplorer.usgs.gov) and preprocessing is performed on SAR images by selecting the desired object which is the house.

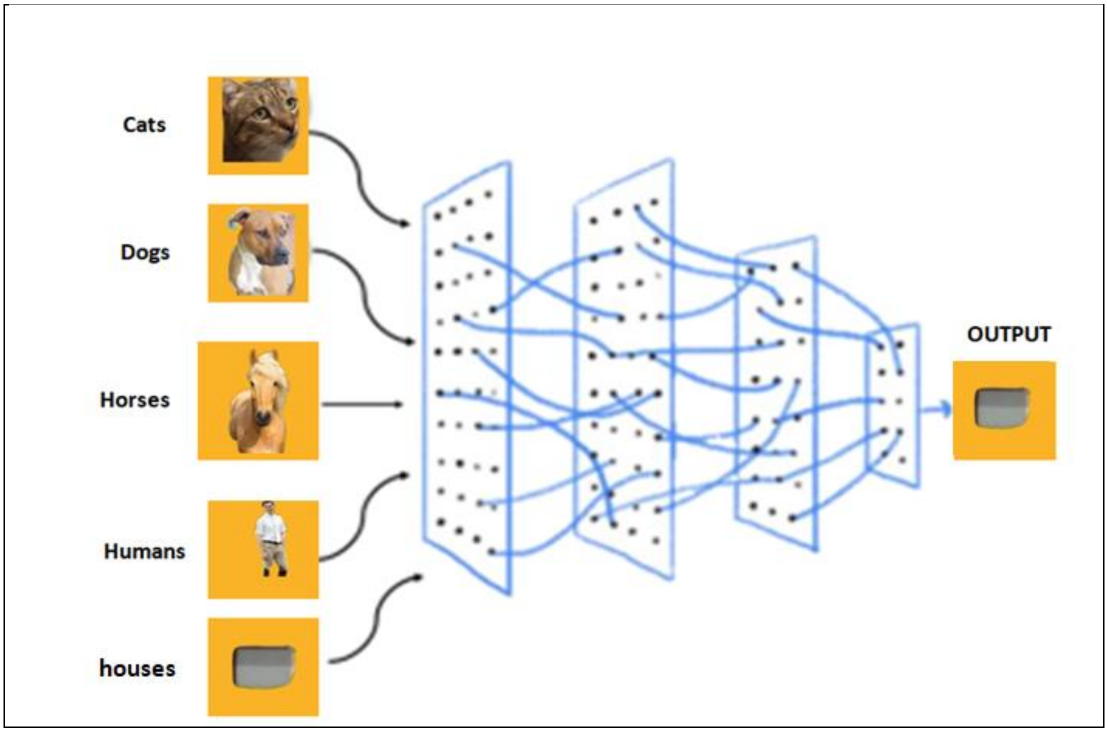


Figure (3) The CNN process towards images in classes.

The first step is to gather the data. It will be the step most complicated and disquieting element. Inputs are also set with the dimensions(224x224) RGB image having a fixed scale. The process of convolution is designed with VGG16, as it generates powerful convolution neural networks. Fortunately, Kaggle named the images that we can download easily.

After import, all the necessary libraries then defining dimensions and locating images. In this step, are defining the dimensions of the image. The best dimensions to describe image sizes are (224x224). Then a file was constructed that will be used to convert all image pixels to their corresponding number and store them in a file in a specific path. Based on the number of classifications and how many images per classification that consuming time in the process. Finally, define the number of epochs and batch sizes.

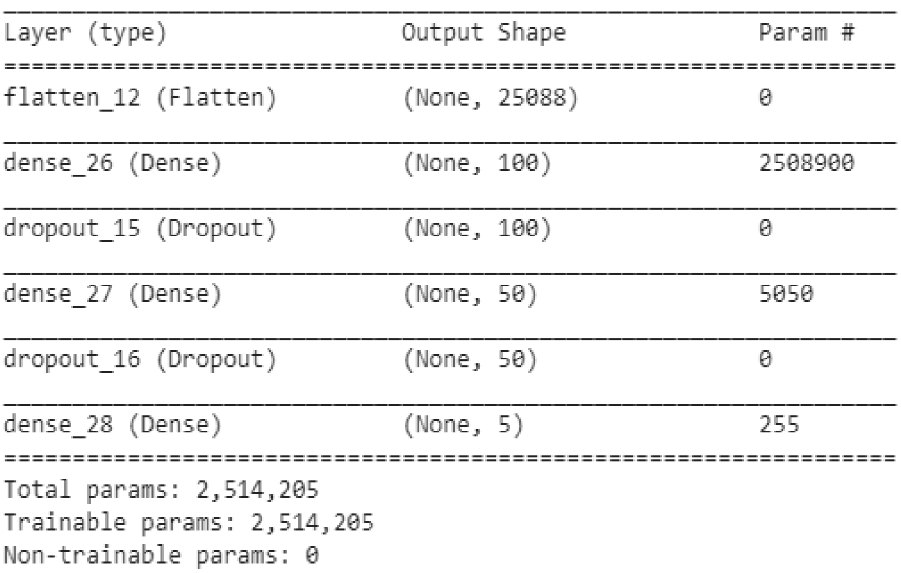
7.2.Download pre-trained model in Keas (VGG16).

Transfer learning is useful because it comes with pre-made neural networks. This step will import the transfer learning aspect of CNN. Each move will integrate CNN's transfer learning feature. Many transfer learning models, such as VGG16, VGG19, ResNet50 are available.In this paper, illustrated using VGG16, which includes only 11 convolutional layers.

For this image classifier operated with five classifications so it didnot take too long to use transfer learning on those images, but concluded that the more images and classifications it will increase the needed time to complete this step, but luckily because only needed to convert pixels of images to numbers once, only needed to do the next step with each training, validation, and testing once-unless happened to remove or to corrupt the image. Then saving files in the program. After that, the weights and features are generated using VGG16.

7.3. Designing and training a CNN model in Keas.

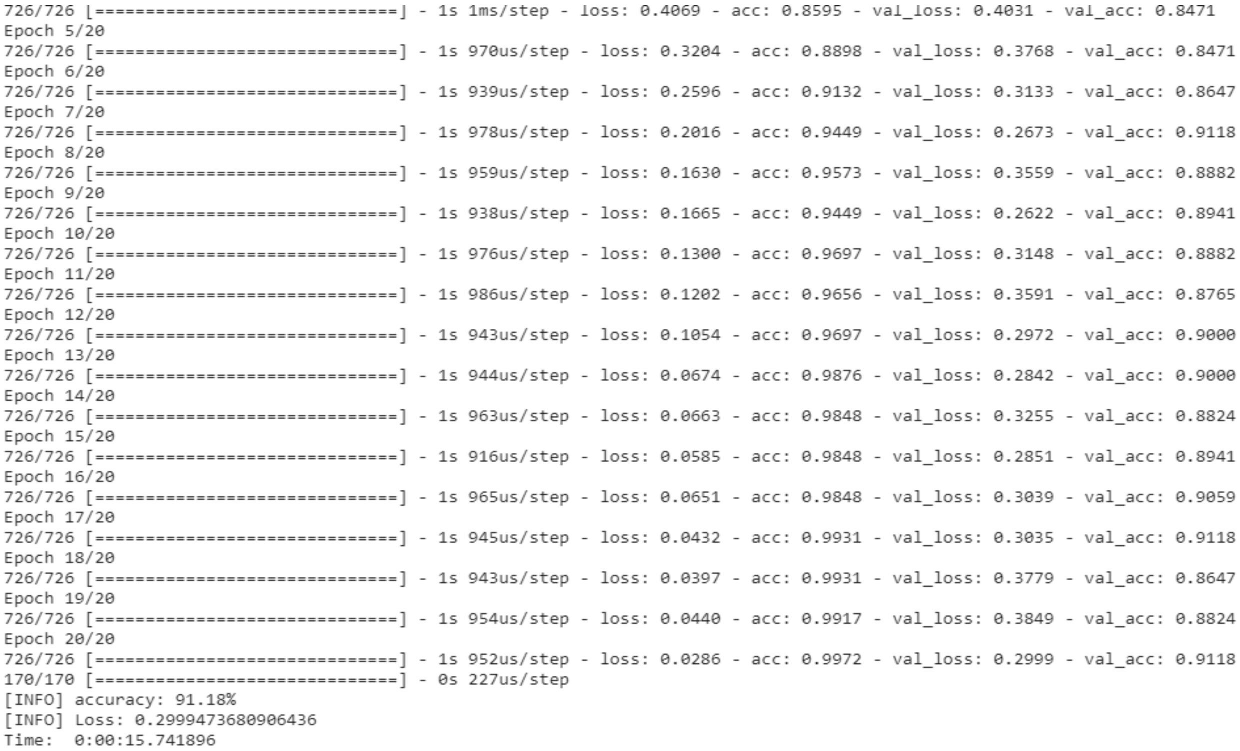
Based on Figure (4), After creating this model the first step is to initialize the configuration with Sequential function. Then flatten the data, and add three more hidden layers.This is a classification labeled as categorical, so the final activation must always be softmax. Finally, build an evaluation phase to test the model of a training set against the validation set for accuracy. As shown in Figure (4) this model now is ready to training the data and then validating it.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
flatten_12 (Flatten)         (None, 25088)             0
_________________________________________________________________
dense_26 (Dense)             (None, 100)               2508900
_________________________________________________________________
dropout_15 (Dropout)         (None, 100)               0
_________________________________________________________________
dense_27 (Dense)             (None, 50)                5050
_________________________________________________________________
dropout_16 (Dropout)         (None, 50)                0
_________________________________________________________________
dense_28 (Dense)             (None, 5)                 255
=================================================================
Total params: 2,514,205
Trainable params: 2,514,205
Non-trainable params: 0
```

Figure (4) CNN Layers of the model

An epoch is how many times the model trains on whole dataset. Each epoch must complete all batches before moving to the next epoch. Too little epoch training will lead to data under fitting and too much can result in data overfitting. The successful training process wanted a loss as small as possible too. Figure (5) show the accuracy and loss of the dataset.

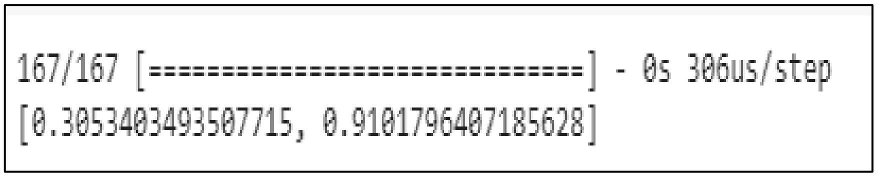


```
726/726 [==============================] - 1s 1ms/step - loss: 0.4069 - acc: 0.8595 - val_loss: 0.4031 - val_acc: 0.8471
Epoch 5/20
726/726 [==============================] - 1s 970us/step - loss: 0.3204 - acc: 0.8898 - val_loss: 0.3768 - val_acc: 0.8471
Epoch 6/20
726/726 [==============================] - 1s 939us/step - loss: 0.2596 - acc: 0.9132 - val_loss: 0.3133 - val_acc: 0.8647
Epoch 7/20
726/726 [==============================] - 1s 978us/step - loss: 0.2016 - acc: 0.9449 - val_loss: 0.2673 - val_acc: 0.9118
Epoch 8/20
726/726 [==============================] - 1s 959us/step - loss: 0.1630 - acc: 0.9573 - val_loss: 0.3559 - val_acc: 0.8882
Epoch 9/20
726/726 [==============================] - 1s 938us/step - loss: 0.1665 - acc: 0.9449 - val_loss: 0.2622 - val_acc: 0.8941
Epoch 10/20
726/726 [==============================] - 1s 976us/step - loss: 0.1300 - acc: 0.9697 - val_loss: 0.3148 - val_acc: 0.8882
Epoch 11/20
726/726 [==============================] - 1s 986us/step - loss: 0.1202 - acc: 0.9656 - val_loss: 0.3591 - val_acc: 0.8765
Epoch 12/20
726/726 [==============================] - 1s 943us/step - loss: 0.1054 - acc: 0.9697 - val_loss: 0.2972 - val_acc: 0.9000
Epoch 13/20
726/726 [==============================] - 1s 944us/step - loss: 0.0674 - acc: 0.9876 - val_loss: 0.2842 - val_acc: 0.9000
Epoch 14/20
726/726 [==============================] - 1s 963us/step - loss: 0.0663 - acc: 0.9848 - val_loss: 0.3255 - val_acc: 0.8824
Epoch 15/20
726/726 [==============================] - 1s 916us/step - loss: 0.0585 - acc: 0.9848 - val_loss: 0.2851 - val_acc: 0.8941
Epoch 16/20
726/726 [==============================] - 1s 965us/step - loss: 0.0651 - acc: 0.9848 - val_loss: 0.3039 - val_acc: 0.9059
Epoch 17/20
726/726 [==============================] - 1s 945us/step - loss: 0.0432 - acc: 0.9931 - val_loss: 0.3035 - val_acc: 0.9118
Epoch 18/20
726/726 [==============================] - 1s 943us/step - loss: 0.0397 - acc: 0.9931 - val_loss: 0.3779 - val_acc: 0.8647
Epoch 19/20
726/726 [==============================] - 1s 954us/step - loss: 0.0440 - acc: 0.9917 - val_loss: 0.3849 - val_acc: 0.8824
Epoch 20/20
726/726 [==============================] - 1s 952us/step - loss: 0.0286 - acc: 0.9972 - val_loss: 0.2999 - val_acc: 0.9118
170/170 [==============================] - 0s 227us/step
[INFO] accuracy: 91.18%
[INFO] Loss: 0.2999473680906436
Time:  0:00:15.741896
```

Figure (5) the accuracy and loss of the dataset.

7.4.Evaluating the model and Predicting the output class of a test image.

Figure (6) shows how well can predict the model just made toward unspecified data. The testing is performed on test data that reached to accuracy (0.910179).



```
167/167 [==============================] - 0s 306us/step
[0.3053403493507715, 0.9101796407185628]
```

Figure (6) evaluating the model.

7.5.Plotting the Loss and Accuracy curve.

The Accuracy curve has great importance in determining the best number of the epoch of iteration to train the dataset to obtain high accuracy instead of assuming random numbers each time. For example in this model random number was assumed for the number of epoch in the first iteration, then after extracting the accuracy curve. The point of intersection of the precision and loss curves will indicate to the best number of epoch in the next training session. Therefore, the best weights and high accuracy are obtained and giving the classification system a correct prediction in most tests. As shown to these graphs in Figures (7) which documented epoch between (2.5 and 5.0) was the strongest as it was the intersection point between accuracy and loss curves.

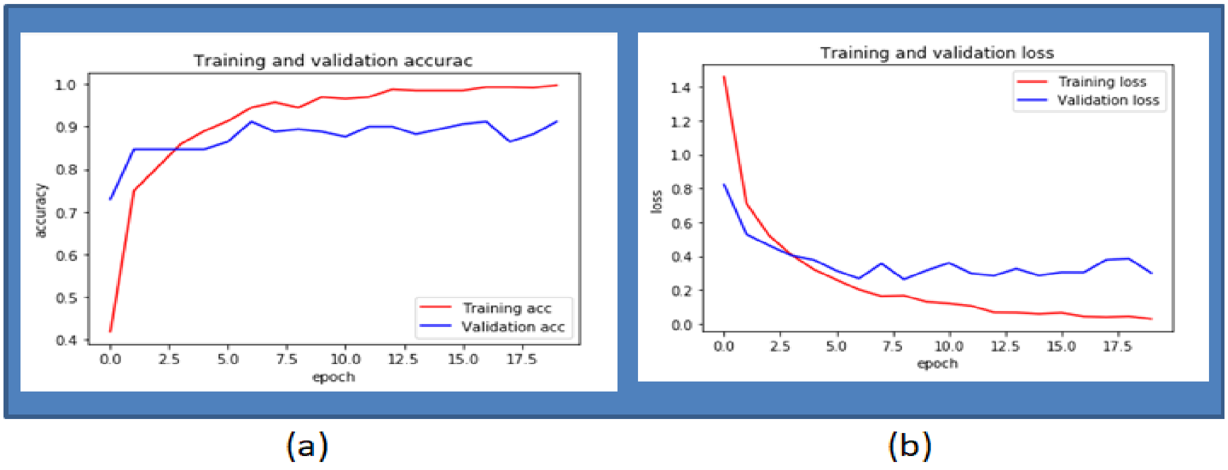
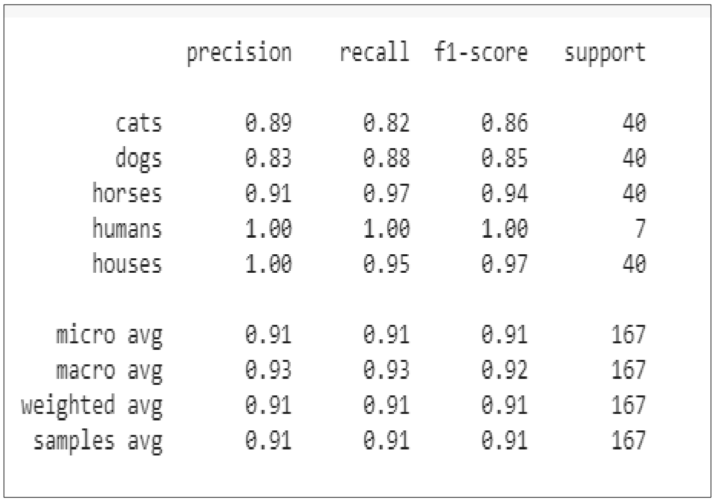


Figure (7.a) training and validation accuracy curve. **Figure (7.**b) training and validation loss curve.

7.6. Plotting the confusion matrix for the result.

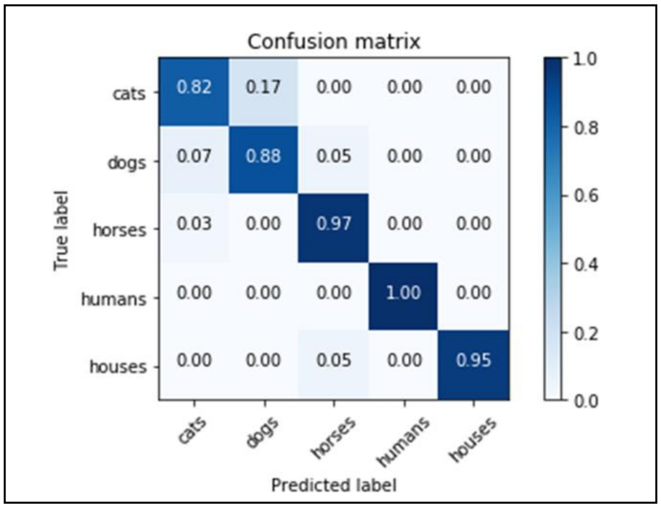
There are two perfect ways to see how well the system can predict or classify. First one is the classification metrics. As shown in Figure (8) and the second way is the confusion matrix as shown in Figure (9) explained that precision and f1-score are major factors here and a higher score means a better pattern.



```
              precision    recall  f1-score   support

        cats       0.89      0.82      0.86        40
        dogs       0.83      0.88      0.85        40
      horses       0.91      0.97      0.94        40
      humans       1.00      1.00      1.00         7
      houses       1.00      0.95      0.97        40

   micro avg       0.91      0.91      0.91       167
   macro avg       0.93      0.93      0.92       167
weighted avg       0.91      0.91      0.91       167
 samples avg       0.91      0.91      0.91       167
```



	cats	dogs	horses	humans	houses
cats	0.82	0.17	0.00	0.00	0.00
dogs	0.07	0.88	0.05	0.00	0.00
horses	0.03	0.00	0.97	0.00	0.00
humans	0.00	0.00	0.00	1.00	0.00
houses	0.00	0.00	0.05	0.00	0.95

Figure (8)the classification metrics for the result. **Figure (9)** the confusion matrix for the result.

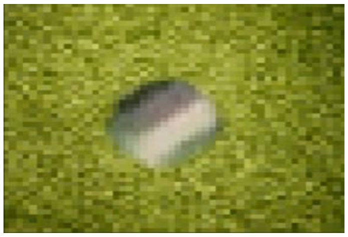
7.7. *Saving and Loading the Trained weights and model.*

Keeping and loading the weights and the trained models. This step enables to keep the trained weights and the trained model in the device so that enable predict the images that want to test and not start from scratch, and this facilitates effort and shortens the test time.

8 RESULTS

After completing the process of training the model some images tested to show the result of the accuracy of each class and explain the probability of accuracy for each class.

Table (2).The result of classification for all classes in percentage

No.	Type of Classes	Name of Classes	Probability of Accuracy %	Tested Image
1.	Non SAR images	Cats	**99.6%**	
2.	Non SAR images	Dogs	0.31%	
3.	Non SAR images	Hourses	0.0%	
4.	Non SAR images	Humans	0.0%	
5.	SAR images	Houses	0.08%	
6.	Non SAR images	Cats	0.51%	
7.	Non SAR images	Dogs	**99.47%**	
8.	Non SAR images	Hourses	0.0%	
9.	Non SAR images	Humans	0.2%	
10.	SAR images	Houses	0.0%	
11.	Non SAR images	Cats	0.0%	
12.	Non SAR images	Dogs	0.0%	
13.	Non SAR images	Hourses	**100.0%**	
14.	Non SAR images	Humans	0.0%	
15.	SAR images	Houses	0.0%	
16.	Non SAR images	Cats	0.0%	
17.	Non SAR images	Dogs	0.0%	
18.	Non SAR images	Hourses	0.0%	
19.	Non SAR images	Humans	**100.0%**	
20.	SAR images	Houses	0.0%	
21.	Non SAR images	Cats	0.0%	
22.	Non SAR images	Dogs	0.0%	
23.	Non SAR images	Hourses	0.0%	
24.	Non SAR images	Humans	0.0%	
25.	SAR images	houses	**100.0%**	

9. DISCUSSION

The result of this paper relies on what aim needs to be accomplished. Those parameters used a convolutional neural network (CNN) to play their roles in assessing the accuracy of image classification. The result was checked by classifying five class types that showed high accuracy to the implementation of the image classification system using CNN. This happened because of the amount of the data that was used, and that includes of 4 classes that were configured and downloaded from Kaggle and a class for SAR images (house class) that we have configured for their limitations, the preparing to house class start by capture the images, then extract the desired objects to be expected and employed as posters ready for training, verification, and testing in the system. Also, the data set used to train the model. Of course, CNN worked fantastically because there was a lot of data. In addition to its ability to predict images captured with two different types of Sensors of Remote Sensing Passive Sensors (such as a photographic camera for example cats class), Active Sensors (such as SAR for example houses class) both of which achieved the possibility of high accuracy in predicting them. If the system was validated with images that were not used during the training model and were not in one of the categories of the system dataset, errors would occur after the classification has been carried out. The errors mean that they cannot recognize untrained images by untrained systems. Finally, when the model size becomes smaller, the training session takes a short time to complete, but the accuracy ratio may be slightly lower compared to the large size of the training model. As for the model used in this paper shown in Figure (5) that Clarified the training session lasted 15.74 seconds, and the accuracy is 91.18%, thus this model success by its speed in training and reaching high accuracy.

10. CONCLUSIONS

To conclude, this research for image classification and overcome the difficulties of CNN's deep training resulting from dataset consist of limited SAR images by using a pre-trained model VGG16 for transfer learning which works as features extractor and it is an effective way to solve the problem of those who need to apply prediction from a limited dataset, then using CNN as a classifier. This target been has achieved where all of the collected tests showed very promising results. CNN technology has been applied in greater detail starting with the collection of training models until image classification. CNN period positions were capable of monitoring precision and avoiding any problems including overproduction. Implementing CNN using the TensorFlow system has also provided good results, as it can simulate, train, and classify the model until it becomes a trained model. This method has shortened a lot of effort and time in the classification of images instead of building a Convolutional Neural Network (CNN) from scratch that needs a lot of tuning and testing.

REFERENCES

Abbas, R. F. (**2020**).A Proposed Approach to Determine the Edges in SAR images. *Iraqi Journal of Science*, 185-192.

Acharya, U. R., Oh, S. L., Hagiwara, Y., Tan, J. H., Adam, M., Gertych, A., & San Tan, R. (**2017**).A deep convolutional neural network model to classify heartbeats. *Computers in Biology and Medicine,* 89, 389-396.

Ahmed, H. M., & Mahmoud, H. H. (**2018**).Effect of Successive Convolution Layers to Detect Gender. *Iraqi Journal of Science,* 1717-1732. Retrieved from http://scbaghdad.edu.iq/eijs/index.php/eijs/article/view/360

Baratloo, A., Hosseini, M., Negida, A., & El Ashal, G. (**2015**).Simple definition and calculation of accuracy, sensitivity and specificity,*Emergency*, 3(2), 48-49.

Chen, S., Wang, H., Xu, F., & Jin, Y. Q. (**2016**).Target Classification using the Deep Convolutional Networks for SAR Images.*IEEE Trans. Geosci. Remote Sens,* 54, 4806–4817.

Chen, S.; Wang, H. (**2014).***SAR Target Recognition Based on Deep Learning.*In Proceedings of the 2014 International Conference on Data Science and Advanced Analytics (DSAA), Shanghai, Chinapp, 541–547.

Christodoulidis, S.,Anthimopoulos, M., Ebner, L., Christe, A., Mougiakakou, S.(**2016**). Multisource Transfer Learning With Convolutional Neural Networks for Lung Pattern Analysis.*IEEE J. Biomed. Health Inform*. 21, 76–84.

Golob, T. F., & Regan, A. C. (**2001**).Impacts of information technology on personal travel and commercial vehicle operations: research challenges and opportunities.*Transportation Research Part C: Emerging Technologies,* 9(2), 87-121.

Gregor, K., Danihelka, I., Graves, A., Rezende, D. J., &Wierstra, D. (**2015**). *DRAW: A Recurrent Neural Network For Image Generation.* 1502- 04623. https://doi.org/10.1038/nature14236

Hosny, A., Parmar, C., Quackenbush, J., Schwartz, L. H., &Aerts, H. J. (**2018**).Artificial intelligence in radiology. *Nature Reviews Cancer* 18(8), 500-510.

Huang, Z., Pan, Z., & Lei, B. (**2017**).Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data. *Remote Sensing* 9(9), 907.

Jain, A. K., Nandakumar, K., & Ross, A. (**2016**).50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79, 80-105.

Kamavisdar, P., Saluja, S., &Agrawal, S. (**2013**).A survey on image classification approaches and techniques.*International Journal of Advanced Research in Computer and Communication Engineering*,2(1),1005–1009. https://doi.org/10.23883/IJRTER.2017.3033.XTS7Z

Korytkowski, M., Rutkowski, L., & Scherer, R. (**2016**). Fast image classification by boosting fuzzy classifiers. *InformationSciences*,327,175–182. https://doi.org/10.1016/j.ins.2015.08.030

Lai, K., & Fox, D. (**2010**). Object recognition in 3D point clouds using web data and domain adaptation. *International Journal of Robotics Research*, 29(8), 1019-1037.

Li, F. F., VanRullen, R., Koch, C., &Perona, P. (**2002**).*Rapid natural scene categorization in the near absence of attention.* Proceedings of the National Academy of Sciences 99 (14), 9596-9601.

Morgan, D.A.(**2015**).*Deep Convolutional Neural Networks for ATR from SAR Imagery*.In Proceedings of the Algorithms for Synthetic Aperture Radar Imagery XXII, Baltimore, 9475-94750. International Society for Optics and Photonics.

Oquab, M., Bottou, L., Laptev, I., &Sivic,J. (**2014**). Learning and Transferring Mid-Level Image Representations Using Convolutional Neural Networks.In Proceedings of the 2014 *IEEE Conference on Computer Vision and Pattern Recognition*, 1717–1724.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., & Swami, A. (**2016**).The limitations of deep learning in adversarial settings.*IEEE European symposium on security and privacy* (EuroS& P),372-387.

Pasolli, E., Melgani, F., Tuia, D., Pacifici, F., & Emery, W. J. (**2013**). SVM active learning approach for image classification using spatial information.*IEEE Transactions on Geoscience and Remote Sensing*, 52(4), 2217–2223. https://doi.org/10.1109/TGRS.2013.2258676

Patterson J, Gibson A.(**2017**). *Deep learning: A practitioner's approach.* O'Reilly Media, Inc.

Rastegari, M., Ordonez, V., Redmon, J., &Farhadi, A. (**2016**).XNOR-net: Imagenet classification using binary convolutional neural networks.*Lecture Notes in Computer Science* (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9908 LNCS, 525–542. https://doi.org/10.1007/978-3-319-46493-0_32

Shin, H.; Roth, H.R.; Gao, M.; Lu, L.; Xu, Z.; Nogues, I.; Yao, J.; Mollura, D.J.; Summers, R.M. (**2016**). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning.*IEEETrans. Med. Image*, 35, 1285–1298.

Weimer, D., Scholz-Reiter, B., &Shpitalni, M. (**2016**).Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. *CIRP Annals* 65(1), 417-420.