Measurement: Sensors 27 (2023) 100775

Contents lists available at ScienceDirect

Measurement: Sensors

journal homepage: www.sciencedirect.com/journal/measurement-sensors

Satellite images for roads using transfer learning

Hussein Ali Al-Iiedane [a,*], Ans Ibrahim Mahameed [b]

[a] Basrah University for Oil and Gas, Basra, Iraq
[b] Aliraqia University, Iraq

ARTICLE INFO

Keywords:
Transfer learning
Image satellite
Convolutional neural networks (CNNs)
Road extraction

ABSTRACT

The challenge of identifying roads in very high resolution (VHR) satellite pictures is a major one in remote sensing. The purpose of this piece was to develop a model for automatically locating and labeling roads on a map as roads or non-roads. As originally conceived, this paradigm was to be a software tool. Hand-sort the satellite images. Automated separation is recommended over manual inspection because of the potential for human error. Together, a convolutional neural network and transfer learning were used to create the model. With a 97.6% success rate for MobileNetv2 and an 81.6% success rate for Alex's Net, the authors were able to identify photos using transfer learning. Given that each point in the spatial feature can now make a reference to any additional contextual data, the accuracy of the road segmentation is enhanced. Our models outperform the state-of-the-art models that have been previously published on the topic in the official Deep Learning Challenge. Our developed model also has a shorter training convergence time than the competing methods. The authors also offer empirical assessments of the best and worst times to deploy non-local blocks in the baseline model. Theoretically, a previously unknown best alternative identification approach was discovered as a result of the methods employed to train the model. The use of transfer learning on a rather sizable dataset allowed for this success. Yet another benefit of the design is that it incorporates data augmentation into its development.

1. Introduction

The challenge of identifying roads in very high resolution (VHR) satellite pictures is a major one in remote sensing. The purpose of this piece was to develop a model for automatically locating and labeling roads on a map as roads or non-roads. As originally conceived, this paradigm was to be a software tool. Hand-sort the satellite images. Automated separation is recommended over manual inspection because of the potential for human error. Together, a convolutional neural network and transfer learning were used to create the model. With a 976% success rate for MobileNetv2 and an 816% success rate for Alex's Net, the authors were able to identify photos using transfer learning. Given that each point in the spatial feature can now make a reference to any additional contextual data, the accuracy of the road segmentation is enhanced. Our models outperform the state-of-the-art models that have been previously published on the topic in the official Deep Learning Challenge [1–4]. Our developed model also has a shorter training convergence time than the competing methods. The authors also offer empirical assessments of the best and worst times to deploy non-local blocks in the baseline model. Theoretically, a previously unknown best alternative identification approach was discovered as a result of the methods employed to train the model. The use of transfer learning on a rather sizable dataset allowed for this success. Yet another benefit of the design is that it incorporates data augmentation into its development [5–9].

The fundamental problem was that, even with dilated convolution, it was challenging to capture long-range correlations. Since local characteristics may be hidden by extra obstructions like shadows, clouds, trees, or buildings because satellite data is collected from above, the single convolution layer, depending on its kernel size, only considers a small number of neighborhoods. Additionally, a deeper CNN design does not necessarily ensure a larger effective receptive field [10–12], neighbourhoods address this issue, we suggest the Transfer learning network in this research. Non-local neural processes calculate the feature map values as a weighted sum of the learning network over all locations. Consequently, it makes it possible for the model to effectively account for long-range dependencies and capture distant information. The contributions of this study are summarized above. When taking part in the DeepGlobe Challenge, our model outperformed any public model that has previously been released with no sophisticated post-processing.

* Corresponding author.
E-mail addresses: Hussein.aliedane@buog.edu.iq (H.A. Al-Iiedane), ans.alkasaab@aliraqia.edu.iq (A.I. Mahameed).

https://doi.org/10.1016/j.measen.2023.100775
Received 31 January 2023; Received in revised form 3 April 2023; Accepted 4 May 2023
Available online 24 May 2023

Transfer learning even surpassed within the official DeepGlobe Challenge, solution [13] was top-ranked [14] while utilizing 43% fewer parameters, fewer GFLOPs, and a quicker training convergence time.

The structure of the paper is as follows: the first section is an introduction of the problem and contributions. The second part is the overview of transfer learning. The detail of the Transfer learning applied for road extraction are in the third section. The experimental result and discussion and displayed in the fourth section. The last section is the conclusion.

1.1. Transfer learning overview

Deep learning is an approach to machine learning that uses artificial neural networks and complex algorithms to train computers to learn from experience, to categorize data, and to recognize images in ways that are strikingly similar to those of the human brain. The use of deep learning to examine massive datasets has recently increased in popularity. Object and image identification are two common applications of convolutional neural networks (CNNs), a form of artificial neural network used in deep learning [15–20]. Deep Learning's use of a convolutional neural network (CNN) makes it possible for it to recognize objects in photos. It's no secret that CNNs are being put to work in a broad variety of fields and applications, from localization and segmentation in computer vision to video analysis and obstacle detection in self-driving cars.

The primary objective of this study is to provide an end-to-end solution for satellite picture recognition that is based on deep learning. For the classification of satellite pictures, we take advantage of transfer learning to enhance state-of-the-art, deep CNN architectures pre-trained on CNN MobileNetv2 and CNN Alex Net. In order to understand specific qualities of the satellite image, this method employs previously gathered information to fine-tune models.

The term "transfer learning" (TL) refers to a type of machine learning in which insights gained from one context are used to the solution of a similar problem. A large number of deep neural network-based transfer learning algorithms have been developed and deployed recently in a wide range of applications, including computer vision, reinforcement learning, and natural language processing [21] [22–24]. Through the use of transfer learning, we improved the performance of many commonly used deep CNN architectures that had been pre-trained on the Mobile Net dataset in order to address the road categorization problem. Transfer learning is an appropriate strategy for this application because the applied dataset is smaller than Mobile Net's. An eight-output unit receives data from a single fully connected layer that uses a feature extractor derived from previously trained models. The feature extractor can use the information it has gleaned from satellite images to better understand more complex characteristics after some adjustment. When compared to building the model from scratch, this greatly simplifies and accelerates the training process [25,26].

CNN have issues when they delve further. This is because the distance between the input layer and the output layer (and the gradient in the opposite direction) increases to the point where information may become lost before crossing to the other side.

2. Applicing knowledge for road extraction

Aerial images are frequently obstructed by other objects since they are taken from extremely high altitudes. The likelihood that a route will be obstructed by another obstruction, such as the shadows cast by high buildings or roads covered by trees, is highest. Capturing long-range relationships is crucial for overcoming this problem, but almost all CNN approaches have trouble doing so. It is challenging to refer to distant information when using a convolution technique because it only refers to local information through a small kernel. Another local method, the recurrent operation, solely uses features from the past and present. Local approaches regularly generate these actions, but they have several disadvantages, including a lower memory and computation efficiency for the model optimization.

Two models pre-trained on the Mobile Net and the AlexNet are evaluated for use in transfer learning. For the classifier, the authors employ a stochastic gradient descent optimizer with weight initialization, setting the momentum to 0.9 [27]. This model is trained at a 32-minibatch size on a single CPU. When the model converges at 25 epochs, training is stopped. The training process begins with a learning rate of 0.001, and the rate is increased by 0.1 after every 7 epochs until convergence is reached. The authors use data augmentation techniques during training to make do with a finite amount of data, hence facilitating an efficient simplification of the model for use with fresh cases. Common image transformations are accounted for in the training set, such as a 60-degree rotation and random horizontal and vertical flips. At this stage, images are reduced in size from their original resolution to 180×180 for the Mobile Net model and 224×224 for the AlexNet model. The rest of the image has been shrunk so that it fits into a 64×64 pixel grid. Categorical cross-entropy is used as a loss function while Softmax is used as an activation function at the classifier's output layer. It could present data contrasting the efficacy of two models (see Fig. 1).

A mobile CNN. Based on an inverted residual structure with residual connections between bottleneck layers, lightweight depth-wise convolutions are used in intermediate expansion layer filters. 19 bottleneck layers follow MobileNetV2's 32-filter fully convolutional layer. The dataset was divided into training and testing subgroups. Training set had 6560 images (80%) while test set had 1640 (20%). Fig. 2-like structure and layers.

AlexNet may be trained on eight separate layers. Except for the output layer, the model consists of five layers, each of which combines fully connected layers with max pooling and Relu activation [28–37]. This technique split the dataset into two parts: training and testing. Of the total number of photos, 6560 (or 80%) were used for training, whereas only 1640 (20%) were used for testing. The layers and structure looked like Fig. 3.

First, after the pre-activation structure of the activation function, the convolutional layer is simpler to train and performs better in terms of generalization. To achieve super-resolution images, a CNN jump-connected network was used. The block unit used 3×3 convolution to create large sensory fields, and 1×1 convolution deepened the network to learn robust feature representations. The network was deepened using simple filters with few parameters to make the most of the hierarchical feature deepening structure. The network outperformed a new algorithm with robust fitting capabilities. Second, before BN and activation work, the convolutional layer does not employ pre-activated dense units. Third, before the activation function and BN, the convolutional layer transfers BN-normalized data to the non-saturated zone so that ReLU can control the degree of activation saturation. If the order of BN and ReLU is changed, however, the results may be unpredictable. ReLU will result in the deactivation of some BN neurons, which leads to BN instability and impairs the performance of the model. In practice, various application scenarios will play different roles in the hierarchy. Fourth, the convolutional layer comes before the activation function, and BN is more suited for denoising, but it causes slow super-resolution convergence and significant loss function volatility, adding to the memory and processing demands.

3. Experiments results and discussions

A. Dataset and Evaluation Metric

They conducted the tests using the DeepGlobe 2018 Road Extraction Challenge dataset [16], which allowed us to evaluate the effectiveness of road extraction. The dataset contains 512×512 pixels wide and tall photos. The satellite of DigitalGlobe acquired With a 0.7 ground sampling distance, each image is presented as an RGB image (GSD). The mask is an identically sized grayscale binary image as the input image.

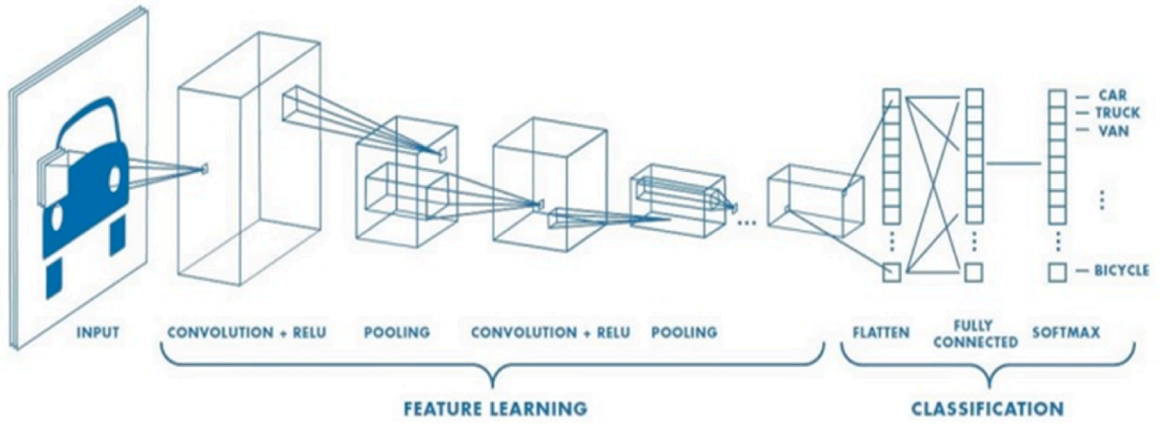


Fig. 1. CNN pathway diagram [23].

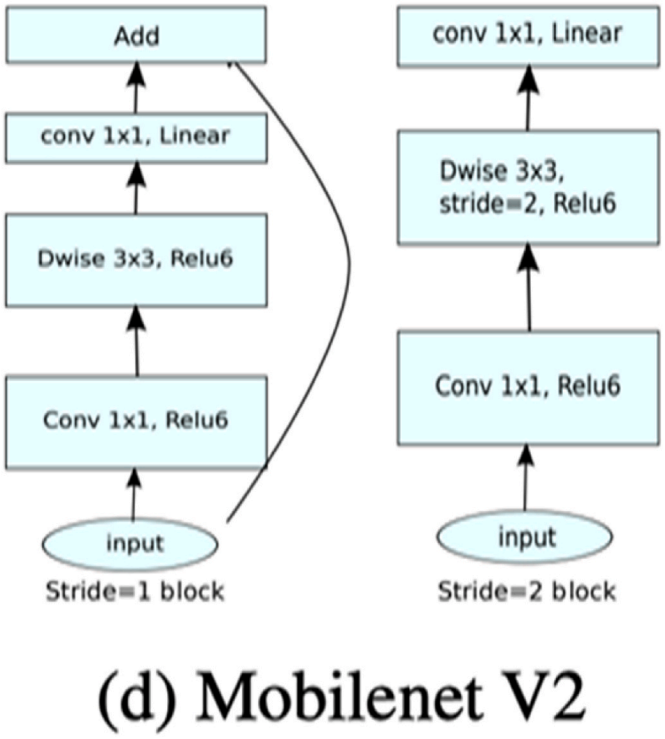


Fig. 2. MobileNetV2 [25].

In the mask illustration, the white represents the highways and the black represents the backdrops.

The dataset contains a total of 8200 images, with 6560 photos designated for training, 820 images designated for validation, and 820 images designated for testing. As the evaluation metric, the author chose to make use of the mean intersection-over-union (mIOU) ratio. DeepGlobe 2018 Road Extraction uses the exact same evaluation metrics as DeepGlobe 2017 Road Extraction. Some examples of these metrics are accuracy and mean square error. In order to properly analyze the models, the pixels that represent roads and those that do not represent roads are respectively labeled with 255 and 0.Pre-processing.

The goal of the image pre-processing stage is to remove any undesirable twists from the image and shrink and normalize it for subsequent processing. According to the need for model development, there are many picture pre-processing techniques found in the prior literature. The most often utilized methods among these are image scaling, image normalization, and covert level to categorical. Using the "Pillow 2.7+ +" Footnote3 Python library, photos were resized for this investigation to guarantee the same size and pixel. In this study, image dimensions of (512 × 512) are taken into account. Additionally, we can alter the pixel intensity with image normalization to make the image appear more realistic. Typically, the majority of image pixels integrate values in the range of 0–255. However, because of how networks are built, it is preferable to execute all values between 0 and 1, since this will be a good fit for model creation. By using this method, the computational complexity of the model training process is reduced.

Images were adjusted, though, using Eq. (1).

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

where X_{min} and X_{max}refer to the minimum and maximum pixel values.

A. Implementation details

Python 3.7 software-associated packages were used to implement the suggested transfer learning classification model. It utilized a 64-bit Windows operating system and an Intel(R) Core(TM) i7-4600 M CPU @ 2.90 GHz processor with 16 GB of main memory and 4 GB of NVIDIA GeForce 940MX graphics. Finding out whether or not an image is labeled as a road is the goal of road extraction identification. Data were divided into three categories before model building: training (80%), validation (10%), and testing (10%), which was trained across 50 epochs. Fig. 3 proposes the used mechanism for route prediction. A confusion matrix can be used to summarize the results of samples of each category that were correctly and wrongly identified. Based on the confusion matrix, they can determine the accuracy (Eq. (2)). Precision, recall, F1-measure, and G-Mean were used to calculate the performance of the classification model. The proportion of occasions for which a prediction was made accurately is known as accuracy. In spite of this, accuracy cannot differentiate between the numbers of correctly categorized samples of each class. This is especially true for the positive class in classification problems. There is a possibility that a reliable classifier made an error and incorrectly categorized some positivity classes as negative. Therefore, precision alone is not necessarily sufficient to determine whether or not a model is useful in solving categorization problems.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

The proportion of accurate forecasts for the favorable class and recall illustrates how well a classifier can distinguish between positive and negative cases. G-Mean is employed to evaluate the two-class recall balance. Since the classification problem now frequently uses this strategy, the G-Mean value will be lower if the model is strongly favoring

Model: "sequential_2"

Layer (type)	Output Shape	Param #
conv2d_10 (Conv2D)	(None, 55, 55, 96)	34944
activation_18 (Activation)	(None, 55, 55, 96)	0
max_pooling2d_6 (MaxPooling 2D)	(None, 27, 27, 96)	0
batch_normalization_16	(None, 27, 27, 96)	384
conv2d_11 (Conv2D)	(None, 17, 17, 256)	2973952
activation_19 (Activation)	(None, 17, 17, 256)	0
max_pooling2d_7 (MaxPooling 2D)	(None, 8, 8, 256)	0
batch_normalization_17 (Bat	(None, 8, 8, 256)	1024
conv2d_12 (Conv2D)	(None, 6, 6, 384)	885120
activation_20 (Activation)	(None, 6, 6, 384)	0
batch_normalization_18	(None, 6, 6, 384)	1536
conv2d_13 (Conv2D)	(None, 4, 4, 384)	1327488
activation_21 (Activation)	(None, 4, 4, 384)	0
batch_normalization_19	(None, 4, 4, 384)	1536
conv2d_14 (Conv2D)	(None, 2, 2, 256)	884992
activation_22 (Activation)	(None, 2, 2, 256)	0
max_pooling2d_8 (MaxPooling 2D)	(None, 1, 1, 256)	0
batch_normalization_20	(None, 1, 1, 256)	1024
flatten_2 (Flatten)	(None, 256)	0
dense_8 (Dense)	(None, 1000)	257000
activation_23 (Activation)	(None, 1000)	0
dropout_6 (Dropout)	(None, 1000)	0
batch_normalization_21	(None, 1000)	4000
dense_9 (Dense)	(None, 512)	512512
activation_24 (Activation)	(None, 512)	0
dropout_7 (Dropout)	(None, 512)	0
batch_normalization_22	(None, 512)	2048
dense_10 (Dense)	(None, 128)	65664
activation_25 (Activation)	(None, 128)	0
dropout_8 (Dropout)	(None, 128)	0
batch_normalization_23	(None, 128)	512
dense_11 (Dense)	(None, 5)	645
activation_26 (Activation)	(None, 5)	0

Total params: 6,954,381
Trainable params: 6,948,349
Non-trainable params: 6,032

AlexNet model. Summary.

Fig. 3. AlexNet model. Summary.

one certain class. F-measure and G-mean were used to evaluate the model's efficiency.

B. Performance Evaluation

For 50 epochs, the CNN model built on Mobile Net was trained to categorize the satellite images. On grayscale photos, each pre-trained model was honed. Fig. 4a and b displays both MobileNet2 and DenseNet-121 CNN model accuracy and loss graph, respectively. Grayscale test images were used to assess DenseNet-121's performance. Table 1 and 2 displays the test data's overall correctness. For thorough performance analysis, Table 3 shows the DenseNet-121 values for precision, recall, and f1-score, using the grayscale test dataset.

While the applied DenseNet-121 model employed a rather large dataset, the vast majority of current techniques used a tiny dataset. By using several state-of-the-art methods, many images and procedures for validation have been developed. Table 3 lists the sample size and validation strategy that each author employed. Table 3 shows that a sizable number of models were developed, tested, and found to successfully categorize traffic occurrences. Some models for multi-class classification achieved an accuracy of up to 95% or more [11], [38–43], despite being more complex and expensive in terms of computation. A meaningful comparison of performance evaluation results and validation procedures is not possible due to the diversity of data sources. However, it is important to note that the 8200 road and non-road picture dataset was used to demonstrate the efficiency of the employed technique. The alternative methods employ a little bit fewer road photos. In an unbalanced dataset where only 219 and 127 photos were used for model development, the methods of Das, Kumar, and Ammor [11] achieved

Table 1
CNN MobileNet result.

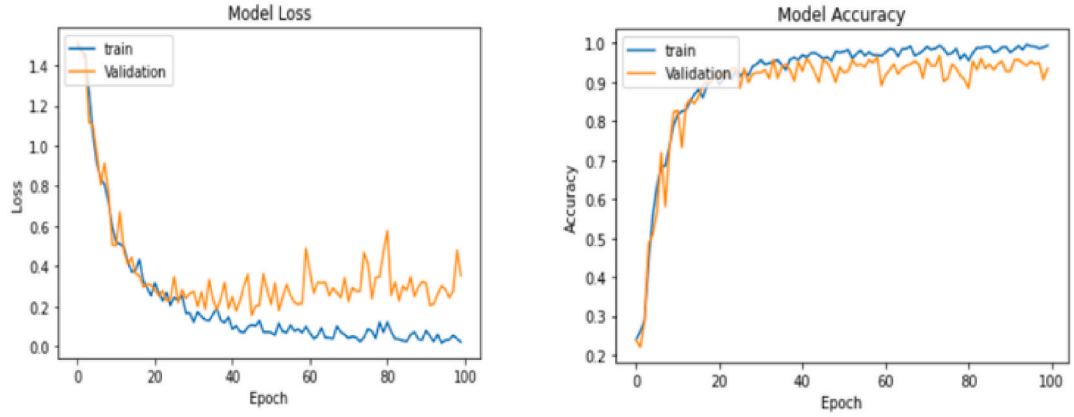
Report of Classification:			
	precision	recall	f1-score
road	1	0.94	0.94
Non road	0.98	0.97	0.978
accuracy			**0.97.6**
macro avg	0.98	0.95	0.96
weighted avg	0.98	0.98	0.98

Table 2
Result CNN DenseNet-121.

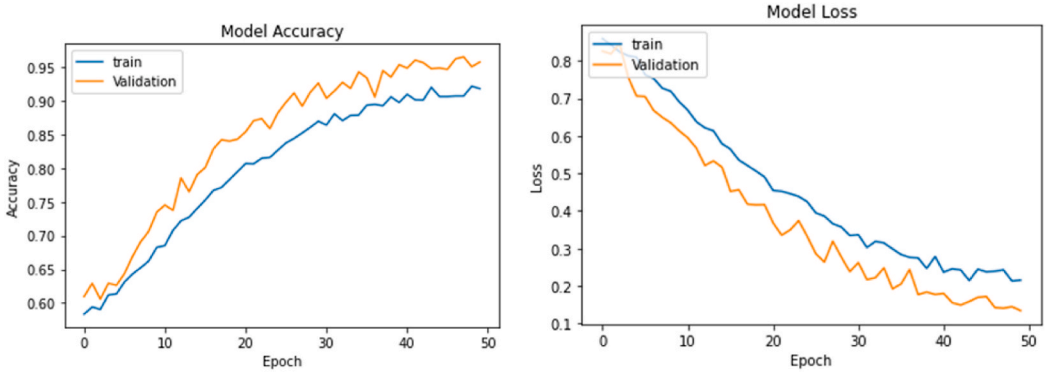
Report of Classification:			
	precision	recall	f1-score
road	81	0.84	0.84
Non road	0.8	0.79	0.78
accuracy			**0.81.6**
macro avg	0.84	0.85	0.82
weighted avg	0.85	0.84	0.81

97.2% and 97.4% overall testing accuracy, respectively. Pathak, Shukla [40] and Shaban, Rabie [34] employed CT scans simultaneously and attained 93% accuracy using a limited picture dataset. Other research [10,12,22,23,35–37], proposed various methods for early identification of images revealing an accuracy rate of less than or equal to 90%.

Models vary in accuracy by a few percent points, on average. We have evaluated all three methods to determine specific knowledge. We



a: Result CNN MobileNetv2



b Result DenseNet-121 CNN

Result CNN MobileNetv2 , DenseNet-121 CNN

Fig. 4. Result CNN MobileNetv2, DenseNet-121 CNN.

Table 3
Model summary of DenseNet.

Layer (type)	Output shape	Param
Model: "model_1"		
input_2 (Input Layer)	(None, 512, 64, 3)	0
conv2d_1 (Conv 2D)	(None, 512, 64, 3)	84
Densenet-121 (Model)	Multiple	6560
global_average_pooling-2d_1	(None, 1024)	0
batch_normalization_1	(None, 1024)	4096
dropout_1 (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 256)	262,400
batch_normalization_2 (Batch	(None, 256)	1024
dropout_2 (Dropout)	(None, 256)	0
root (Dense)	(None, 2)	524

maintain a close check on every aspect of publicly available models, monitoring their accuracy at every level, from the lowest Level Alex Net accuracy of 81.6% all the way up to the highest Rate of Departure from Mobile Net accuracy of 97.0%.

Exciting new research has been conducted in the area of automating the classification of satellite images of the periphery, which has applications in the detection of different types of roads for the purpose of extraction. Early attempts to tackle the problem relied on analysis and manual feature extraction using algorithms from standard machine learning. In recent years, deep learning techniques, and convolutional neural networks in particular, have been frequently applied in this direction. To address this issue, this study employs transfer learning that has been pre-trained on the MobileNet v2. The authors examine and contrast the two models as a consequence of their investigation. Model availability is higher for mobilenet v2 (96.6%) than DenseNet-121 (81.6%). A single incorrectly labeled instance can render a whole data load incomplete; the sheer volume of the dataset is another possible cause. However, low-resolution cameras and pixels still allow for some interesting results. To some extent, the wide performance range may be explained by the care with which we chose our hyper parameters and the weights at which to start. In addition, we find that the models' overall performance does not change dramatically even when we use a variety of designs with different combinations of depth, number of layers, parameters, or branches.

4. Conclusions

In this research, the authors proposed the transfer learning model for satellite images of road extraction. All of the features in the satellite images are referenced in the operation, which records remote information. More accurately than any other publication, our Transfer learning won the official DeepGlobe 2018 Road Extraction Challenge. With fewer parameters, fewer GFLOPs, and a quicker training convergence time, it also performed better than the top-ranked solution. Our investigation into transfer learning blocks involved rearranging the placements, as well as the paired functions. It is essential for satellite photos to go through an automated image categorization procedure, which is carried out by computer-aided processing systems. The analysis of satellite images is a challenging and time-consuming process. According to the findings of this research, the performance of deep learning-based route extraction from satellite images was superior to that of accuracy. Overall, the recall is correct 94% of the time, and its accuracy rating is 97.6%. The average amount of time required for calculation by the transfer learning model is 195.35 s. The deep learning architecture was utilized for the process of picture classification by our team. For example, multiple different classification methods were utilized to provide support for our assumption.

To get a true view of the infection rate, which correlates closely with daily infected cases, it would be advantageous to boost dependability, as shown by the computational testing approach. The method uses machine learning to eliminate human judgment errors and save time and resources. Our investigation has many shortcomings. Future network architecture and optimization may improve categorization accuracy. Second, certain sites lack training data during the early stages of road extraction. Many images may not be enough for reliable prediction. Lack of Grad Cam representations of the utilized DenseNet-121 model hinders reading. Future research may test additional or modified deep learning techniques to improve road extraction from satellite imagery.

Credit author statement

H. R. Al-NaIedane: Paper written, simulation, english editing, paper revised. **A. M. Mahamood**: references arrangement, out put figures, related work.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Run Ma, Shahab Wahhab Kareem, Ashima Kalra, Rumi Iqbal Doewes, Pankaj Kumar, Shahajan Miah, Optimization of electric automation control model based on artificial intelligence algorithm, Wireless Commun. Mobile Comput. (2022), https://doi.org/10.1155/2022/7762493. Article ID 7762493, 9 pages, 2022.
[2] M. Amin Ali, S. Wahhab Kareem, A.S. Mohammed, "Evaluation of electrocardiogram signals classification using CNN, SVM, and LSTM algorithm: a review,", in: 2022 8th International Engineering Conference on Sustainable Technology and Development (IEC), 2022, pp. 185–191, https://doi.org/10.1109/IEC54822.2022.9807511.
[3] R.O. Duda, P.E. Hart, Use of the Hough transformation to detect lines and curves in pictures, Commun. ACM 15 (1) (1972) 11–15.
[4] H.A. Muhamad, S.W. Kareem, A.S. Mohammed, A comparative evaluation of deep learning methods in automated classification of white blood cell images, in: 2022 8th International Engineering Conference on Sustainable Technology and Development, IEC), 2022, pp. 205–211, https://doi.org/10.1109/IEC54822.2022.9807456.
[5] J. Canny, "A computational approach to edge detection," Pattern Analysis and Machine Intelligence, IEEE Trans. PAMI-8 (1986) 679–698, 12.
[6] F. Tupin, H. Maitre, J.-F. Margin, J.-M. Nicolas, E. Pechersky, Detection of linear features in SAR images: application to road network extraction, IEEE Trans. Geosci. Rem. Sens. 36 (3) (1998) 881–890.
[7] F. Khoshaba, S. Kareem, H. Awla, C. Mohammed, Machine learning algorithms in Bigdata Analysis and its applications: a review, in: 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications, HORA), 2022, pp. 1–8, https://doi.org/10.1109/HORA55278.2022.9799848.
[8] Hoshang Q. Awla, Shahab W. Kareem, Amin S. Mohammed, Bayesian network structure discovery using antlion optimization algorithm, Int. J. Syst. Innov. 7 (1) (2022) 46–65.
[9] Shahab Wahhab Kareem, Mehmet Cudi Okur, Falcon optimization algorithm for bayesian network structure learning, Comput. Sci. 22 (4) (2021) 553–569.
[10] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, 2017 arXiv preprint arXiv:1706.05587.
[11] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder- decoder with atrous separable convolution for semantic image segmentation, in: Proc. ECCV, 2018.
[12] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable convolutional networks, 2017 arXiv preprint arXiv:1703.06211.
[13] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[14] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, Hughes, D. Tuia, R. Raska, Deepglobe, A challenge to parse the earth through satellite images,", in: Proc. CVPR Workshops, 2018, 2018.
[15] F. Hu, G.-S. Xia, J. Hu, L. Zhang, Road extraction using SVM and image segmentation, Photogramm. Eng. Rem. Sens. 70 (12) (2004) 1365–1371.
[16] Y. Wei, Z. Wang, M. Xu, Road structure refined CNN for road extraction in the aerial image, IEEE Geosci. Remote Sens. Lett. 14 (5) (2017) 709–713, https://doi.org/10.1109/LGRS.2017.2672734.
[17] M. Ullah, A. Mohammed, F. Alaya Cheikh, Pednet: a spatiotemporal deep convolutional neural network for pedestrian segmentation, J. Imaging 4 (9) (2018) 107.

[18] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, Y. Bengio, The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation, in: Proc. CVPR Workshops, 2017.
[19] Z. Zhang, Q. Liu, Y. Wang, Road extraction by deep residual u- net, IEEE Geosci. Remote Sens. Lett. 15 (5) (May 2018) 749–753, https://doi.org/10.1109/LGRS.2018.2802944.
[20] L. Zhou, C. Zhang, M. Wu, Linknet: Exploiting encoder representations for efficient semantic segmentation, in: Proc. VCIP, 2017.
[21] H. Wang, Y. Fan, Z. Wang, L. Jiao, B. Schiele, Parameter-free spatial attention network for person re-identification, 2018 arXiv preprint arXiv:1811.12150.
[22] L. Zhou, C. Zhang, W. M, "D-linknet: linknet with pre-trained encoder and dilated convolution for high-resolution satellite imagery road extraction,", in: Proc. CVPR Workshops, 2018.
[23] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, Hughes, D. Tuia, R. Raskar, Deepglobe 2018: a challenge to parse the earth through satellite images, in: Proc. CVPR Workshops, 2018.
[24] H. Zhao, S. Liu, S. Jianping, C.C. Loy, D. Lin, J. Jia, Psanet: Point-wise spatial attention network for scene parsing, in: Proc. ECCV, 2018.
[25] C. Han, F. Shen, L. Liu, Y. Yang, H.T. Shen, Visual spatial attention network for relationship detection, in: Proc. MM, 2018.
[26] H.Q. Awla, A. Rahman Mirza, S.W. Kareem, "An automated CAPTCHA for website protection based on user behavioral model,", in: 2022 8th International Engineering Conference on Sustainable Technology and Development, IEC), 2022, pp. 161–167, https://doi.org/10.1109/IEC54822.2022.9807472.
[27] W. Luo, Y. Li, R. Urtasun, R. Zemel, Understanding the effective receptive field in deep convolutional neural networks, in: Proc. NIPS, 2016.
[28] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proc. CVPR, 2018.
[29] Y. Zhang, K. Li, K. Li, Z. Bineng, Y. Fu, Residual non-local attention networks for image restoration, in: Proceedings of the International Conference on Learning Representations (ICLR), 2019.
[30] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. CVPR, 2016, pp. 770–778.
[31] Roojwan Hawezi, Farah S. Khoshaba, Shahab Kareem, A comparison of automated classification techniques for image processing in video internet of things, Comput. Electr. Eng. 101 (2022), https://doi.org/10.1016/j.compeleceng.2022.108074.
[32] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, in: Proceedings of the 3rd International Conference on Learning Representations (ICLR, 2015.
[33] A. Buades, B. Coll, J.-M. Morel, A non-local algorithm for image denoising, Proc. CVPR (5) (2005) 709–713.
[34] Z. Zhong, J. Li, W. Cui, H. Jiang, Fully convolutional networks for building and road extraction: preliminary Results, in: Proc. IEEE IGARSS, 2016.
[35] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, Springer, Cham, 2012.
[36] J. Qin, et al., A biological image classification method based on improved CNN, Ecol. Inf. 58 (2020), 101093.
[37] G. Huang, et al., Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.
[38] O. Filin, A. Zapara, S. Panchenko, Road detection with eosresunet and post vectorizing algorithm, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2018, pp. 211–215.
[39] T. Sun, Y. Wenxiang, Y. Wang, Stacked u-nets with multi-output for road extraction, in: Proc. CVPR Workshops, 2018.
[40] J. Doshi, Residual inception skip network for binary segmentation, in: Proc. CVPR Workshops, 2018.
[41] S. Roy, I. Kiral-Kornek, S. Harrer, ChronoNet: a deep recurrent neural network for abnormal EEG identification, in: Conference on Artificial Intelligence in Medicine in Europe, Springer, Cham, 2019.
[42] T. Sarkar, A. Hazra, N. Das, Classification of colorectal cancer histology images using image reconstruction and modified DenseNet, in: International Conference on Computational Intelligence in Communications and Business Analytics, Springer, Cham, 2021.
[43] N. Hasan, Y. Bao, A. Shawon, et al., DenseNet convolutional neural networks application for predicting COVID-19 using CT image, SN Comput. Sci. 2 (2021) 389.