Article

CNN Algorithm for Roof Detection and Material Classification in Satellite Images

Jonguk Kim ¹ , Hyansu Bae ², Hyunwoo Kang ³ and Suk Gyu Lee ^{4,*}¹ Korea Polytechnic VII, 51-88 Oedongbanrim-ro, Seongsan-gu, Changwon-si 51518, Korea; dolphin89@kopo.ac.kr² Korea Electrotechnology Research Institute, 12, Jeongiui-gil, Seongsan-gu, Changwon-si 51543, Korea; bhs8017@keri.re.kr³ Korea Polytechnic VI, 15 Gukchaebosang-ro 43-gil, Seo-gu, Daegu 41765, Korea; hwkang@kopo.ac.kr⁴ Department of Electrical Engineering, Yeungnam University, 280 Daehak-ro, Gyeongsan 38541, Korea

* Correspondence: sglee@ynu.ac.kr; Tel.: +82-053-810-2487

Abstract: This paper suggests an algorithm for extracting the location of a building from satellite imagery and using that information to modify the roof content. The materials are determined by measuring the conditions where the building is located and detecting the position of a building in broad satellite images. Depending on the incomplete roof or material, there is a greater possibility of great damage caused by disaster situations or external shocks. To address these problems, we propose an algorithm to detect roofs and classify materials in satellite images. Satellite imaging locates areas where buildings are likely to exist based on roads. Using images of the detected buildings, we classify the material of the roof using a proposed convolutional neural network (CNN) model algorithm consisting of 43 layers. In this paper, we propose a CNN structure to detect areas with buildings in large images and classify roof materials in the detected areas.

Citation: Kim, J.; Bae, H.; Kang, H.; Lee, S.G. CNN Algorithm for Roof Detection and Material Classification in Satellite Images. *Electronics* **2021**, *10*, 1592. <https://doi.org/10.3390/electronics10131592>

Academic Editors: Chimani Kwan and D. J. Lee

Received: 25 May 2021

Accepted: 28 June 2021

Published: 1 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: CNN; deep learning; detect roof; satellite images

1. Introduction

Artificial intelligence algorithms demonstrate outstanding results in predicting or discriminating complex or nonlinear models. Among nonlinear models, disaster-damaged models are difficult to predict because they are affected by a variety of variables, such as weather and ground. Currently, images obtained from satellites are available as well as sufficient computing power to process these images. Satellite images come in three main types depending on the satellite's shape [1]. The first is an image containing color information that collects light from the reflected sun, the second is an image that measures the heat on the surface using infrared rays, and the third is an image that uses water vapor in the upper atmosphere. In previous studies, solely RGB photos were used to classify three types of vegetation land cover (tree, shrub, and grass). DeepLabV3+, a deep learning method for semantic segmentation, was applied [2,3]. In our study, the satellite images containing color information were utilized to classify roofs based on color information. The images of satellites are very large and contain very large amounts of imaging information. To interpret these images, research on image spatial decomposition analysis is increasing. Many images are overlapped to convert satellite images into 3D [4–6]. Since digital surface model (DSM) images [7–10], which contain so much information, are generally lower in resolution than LiDAR data [11–13], research on 3D pointer cloud Easy Matching technology continues to improve with a focus on accuracy [14–17]. Image processing algorithms using satellite images use algorithms such as grid-based [18–20], spectral-based [21–24], and resolution-based [25,26] to classify zones and distinguish required data using differences from surrounding data [27].

Natural disasters, such as earthquakes, hurricanes, and floods, can have fatal effects on people and communities. It is good to predict and defend these hazards, but these

prediction models are expensive and time-consuming because they vary from region to region and include massive quantities of data. In order to prepare for such trouble, buildings may be protected, especially incomplete or poorly constructed buildings that are exposed to greater damage. Roofs were used as surface data to approximately analyze the safety of the building. A roof is a part of the final stage of construction, so it can be judged that a completed building has been completed because the roof is also completed. There are cluster-based [28–30] and object-based [31,32] classification methods to determine roofs on maps. Satellite image processing methods currently being studied are used to supplement the needs of existing data through the fusion of satellite images and required sensor data and are often used to acquire a wide range of approximate information, such as overall distribution and environmental situations, rather than fine-grained information obtained from images. When sensor data in the target location are difficult to collect or additional data are limited, these methods can be problematic. Roof detection is also studying recognition in narrow spaces by utilizing three-dimensional data from areas where buildings are mainly distributed or by recognizing specific objects or patterns that roofs contain. In this paper, we proposed an algorithm to classify satellite images in two-dimensional form and the environment in which the building is mainly distributed and the surrounding information, to classify the location of the building without additional data, and to classify the material of the roof by dividing each building. Building information can be obtained and secured using satellite image information through segmentation and classification. According to these studies using satellite imagery, since architectures are typically elevated objects in images, the height details given by DSM data have the following characteristics:

1. Color: The colors of the roads and roofs of each building are generally distinct, so use the colors to differentiate them.
2. Road: Use occlusion to classify spaces to ensure that buildings and roads are properly classified.
3. Stereo: Using the image's height information, identify the places where the height information varies quickly.
4. Noise: Noise in 3D image data is divided into sections where the structure is imperfect or distinctly defined, such as corners, fault lines, and valleys.
5. Pattern: A completed roof will have a specific pattern. It is best to classify using meaningful information such as color, material, and shape, but it is difficult to find these patterns and forms. The shape and color of the roof of the building vary, and even if it is the roof of the same building, the shape and material can be different, causing difficulties in classification.

The framework of each building is extracted using environmental data around the building, and the roof of the building is verified for the material and completion of the building so that it may be identified in advance in the emergency of a disaster. To determine the roof, we utilize convolutional neural network (CNN) algorithms that detect areas where the roof is likely to be present in images of large satellites, use roof features to detect the roof, and propose the material of the detected roof. The proposed CNN model consists of 43 layers. Learning has been classified into four popular types of roof materials, and since it often exhibits patterns similar to those around it, we designed the model intending to classify features well. The learning results of the proposed model showed an approximately 9% accuracy improvement in material classification compared to GoogleNet. We propose an algorithm to detect roofs from satellite images and learn them to classify the materials of roofs.

2. Roof Detection Using Image Processing

In order to detect the roof in the video, the location of the roof shall be detected. Satellite images are very large in size and high in terms of resolution, so they are processed separately by region.

To detect the roof of a building, first separate the space by road. Because the building is divided by roads, it is detected in the video. The areas in which space is divided by a continuous or unique reference line in the image is determined to be roads and paths because road and roads are built in a straight line in the image. Figure 1 shows the areas separated by the road. In general, buildings are likely to exist in spaces separated by the characteristics of straight roads.

Figure 1. A road-separated area.

These spaces are explored because buildings are likely to exist in areas that are divided by roads. First, look for parts that are of different colors. Land and roofs are usually composed of different colors, so we set up areas where buildings are likely to exist based on colors in the separated areas. Figure 2 shows the separation of space again based on color in places where there are likely to be buildings separated by a lane. When blue is referred to as land, the divided colored space on the ground is considered to be a house. However, since it is difficult to tell which section is the right building, it separates all places where the color is distinct.

Figure 2. Areas separated by colors.

Next, color-coded areas should be detected where the actual building is located. At this time, the satellite image should be composed of a three-dimensional form or the depth map should be extracted from the image using stereo algorithms. In the case of structures, the ground determines the lower area of the color-coded area, with all spaces higher

than the ground being judged as buildings. However, if three-dimensional data cannot be obtained, we determine it by using noise or patterns in the region. Figure 3 shows an image containing a building used to determine an area in an environment without three-dimensional data.

Figure 3. Roof area Image with noise and pattern.

We isolate the noise region using the image in Figure 3. Differentiated locations such as roads and buildings are divided by using Fourier transform because pixel differences occur and regions are divided. To apply Fourier transform to the image [33], we separate the RGB channels and apply them for each channel.

We used Equation (1) to add discrete Fourier transform to images, divided by pixels for Fourier transform. The frequencies are: u in the x -axis direction and v in the y -axis direction, W in the horizontal size of the image, and H in the vertical size of the image.

$$F(u, v) = \frac{1}{WH} \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} f(x, y) e^{-j2\pi(\frac{ux}{W} + \frac{vy}{H})} \quad (1)$$

The results of applying Equation (1) to Figure 3 are shown in Figure 4. We show the roof area by showing noise and patterns. The figure shows that the roof of the building is noisier than the ground, and a particular pattern of the roof appears. The location of these noise patterns is determined by the separation of roofs and houses. Using noise, we detect regions by setting thresholds in the delimited data (R1.2) (Figure 4a) where regions consisting of below thresholds are classified as empty or continuous spaces, such as roads, and pixels with changes in values above thresholds are all boundaries. Figure 4b uses the gradient of the image to find the changing interval. Since it is a two-dimensional figure with x and y data, the x and y axes are added respectively. The changing points can be seen as boundary areas between roads or buildings. Using these data, matching each pixel allows us to check the boundaries of changing pixel values and detect buildings based on the areas where the boundaries appear prominently.

Using the separated data in Figure 4, the edge points of the building are detected using the edge part of the area where the building exists in the original image. To detect corner points in classified data using thresholds, we compared data in eight directions of pixels in that part and used the part with changes in values in three or more proximity pixels as corner points. Figure 5 shows the corner points of the buildings extracted from the image as red dots and divides the area of the roof based on these corner points. We use these divided roof areas as the input image of the learning.

Figure 4. Area classification using environmental information (a) Noise and pattern, (b) gradient image, (c) Data Matching.

Figure 5. Detected roof edge point.

Using the detected corner points, we connected the corner points of the same building and separated them for each building. Because roofs of the same building often consist of the same color, we explored the x and y axes based on the color of a point to create a boundary of the building by connecting the outermost corner points.

3. Classification of the Material of the Roof

The material of the roof is used by the CNN algorithm in the area where the roof is present as detected in Section 2. There are four types of roofing materials: concrete cement, healthy metal, incomplete, and irregular metal. The reason why roofs are divided into four types is that different countries have different architectural styles for different reasons, such as climate and religion, and the materials and shapes of roofs vary depending on the architectural style. We use CNN to learn using a database of roofs that have been designed with each material. Figure 6 shows an example of an image for each material.

Figure 6. Learning materials for each image.

Figure 7 exhibits the structure of the proposed CNN. The proposed structure consists of a total of 12 layers, including convolutional, pooling, and concat layers.

Figure 7. Convolution Configuration Diagram.

Figure 8 displays the initial layer of the proposed structure. In the input image, max pooling is used in the initial 3×3 convolutional layer to highlight the feature values of the initial data. Max pooling is used to differentiate between patterns and materials, and characteristics are best reflected while max pooling only leaves broad values. After that, filters in three sizes were used to isolate different characteristics, and data were merged using concatenation.

Figure 8. Proposed structure of Convolutional Layer1.

The calculated value is divided into two parts. The left-hand side of the structure uses the convolutional layer and the pooling layer to make the features more prominent, while the right-hand side has features close to the initial value to avoid losing the initial features. Table 1 presents the parameters of the layers used in Figure 8. Weights initializer uses the “He method” and filters, pools, strides, and padding values are set as shown in the table. To prevent the size of the data from decreasing, padding was output equal to the size of the image being entered, with a value of 0.

Table 1. Layer Parameters in Figure 8.

Layer	Size (Filter, Pool)	Num Filters	Stride	Padding Value	Data Size	Weights Initializer	Bias Initializer
Input	224,224,3				$224 \times 224 \times 3$		
Conv. Layer1	3,3	64	2,2	0	$112 \times 112 \times 64$	He	0
MaxPooling1	5,5		1,1		$112 \times 112 \times 64$		
Conv. Layer2	3,3	32	2,2	0	$56 \times 56 \times 32$	He	0
Conv. Layer3	5,5	64	2,2	0	$56 \times 56 \times 64$	He	0
Conv. Layer4	3,3	64	2,2	0	$56 \times 56 \times 64$	He	0
Conv. Layer5	7,7	16	2,2	0	$56 \times 56 \times 16$	He	0
Concat1					$56 \times 56 \times 96$		
Concat2					$56 \times 56 \times 80$		

For the left region of Figure 9, we extract the feature points with the structure shown in Figure 10. All of these calculated features are combined into concatenation to classify roofs using a fully connected layer. Table 2 sets the number of filters and pool size filters as the parameters of the layer in Figure 9 and similarly determines the size of the padding so that the image does not become smaller, and the value is entered as 0.

Figure 9. Proposed structure of Convolutional Layer2.

Figure 10. Proposed structure of Convolutional Layer3.

Table 2. Layer Parameters in Figure 9.

Layer	Size (Filter, Pool)	Num Filters	Stride	Padding Value	Data Size	Weights Initializer	Bias Initializer
Conv. Layer6	3,3	64	1,1	0	$56 \times 56 \times 64$	He	0
Max Pooling2	5,5		1,1		$56 \times 112 \times 32$		
Conv. Layer7	5,5	32	1,1	0	$56 \times 56 \times 32$	He	0
Max Pooling3	3,3		1,1		$56 \times 112 \times 32$		
Max Pooling4	3,3		2,2		$28 \times 28 \times 64$		
AVG Pooling5	3,3		2,2		$28 \times 28 \times 96$		
Max Pooling6	5,5		2,2		$28 \times 28 \times 96$		
AVG Pooling7	5,5		2,2		$28 \times 28 \times 96$		
Concat3					$28 \times 28 \times 128$		
Concat4					$28 \times 28 \times 288$		
Max Pooling8	5,5		2,2		$14 \times 14 \times 288$		
Conv. Layer8	3,3	64	4,4		$14 \times 14 \times 64$	He	0
Conv. Layer9	5,5	32	4,4		$14 \times 14 \times 32$	He	0
Concat5					$14 \times 14 \times 96$		
Conv. Layer10	5,5	64	2,2		$7 \times 7 \times 64$	He	0
Conv. Layer11	1,1	16	2,2		$7 \times 7 \times 16$	He	0

Figure 10 creates a structure to highlight various feature points by positioning a convolutional layer at the end of the network to extract feature points from the images.

The later layers changed the filter size to find the features highlighted earlier once again. Table 3 shows the parameters of the layer used in Figure 10.

Table 3. Layer Parameters in Figure 10.

Layer	Size (Filter, Pool)	Num Filters	Stride	Padding Value	Data Size	Weights Initializer	Bias Initializer
Conv. Layer12	3,3	32	1,1	0	$14 \times 14 \times 32$	He	0
AVG Pooling9	5,5		1,1		$14 \times 14 \times 288$		
Conv. Layer13	5,5	32	1,1	0	$14 \times 14 \times 32$	He	0
AVG Pooling10	3,3		1,1		$14 \times 14 \times 288$		
Conv. Layer14	1,1	32	1,1	0	$14 \times 14 \times 32$	He	0
Conv. Layer15	5,5	64	2,2	0	$7 \times 7 \times 64$	He	0
Conv. Layer16	1,1	32	2,2	0	$7 \times 7 \times 32$	He	0
Conv. Layer17	3,3	128	2,2	0	$7 \times 7 \times 128$	He	0
Conv. Layer18	1,1	32	2,2	0	$7 \times 7 \times 32$	He	0
Conv. Layer19	3,3	32	2,2	0	$7 \times 7 \times 32$	He	0
Conv. Layer20	5,5	64	2,2	0	$7 \times 7 \times 64$	He	0
Conv. Layer21	3,3	64	2,2	0	$7 \times 7 \times 64$	He	0
Conv. Layer22	1,1	16	2,2	0	$7 \times 7 \times 16$	He	0
Conv. Layer23	5,5	64	2,2	0	$7 \times 7 \times 64$	He	0
Conv. Layer24	1,1	16	2,2	0	$7 \times 7 \times 16$	He	0
Concat6					$7 \times 7 \times 592$		
Global AVGPooling	7,7				$1 \times 1 \times 592$		

The fully connected layer is used to learn the features from the previously computed image. The number of hidden layers and the number of nodes is a highly experienced part of the user's design phase. We found several cases of avoiding overfitting and increasing learning rates through iterative learning of the number of layers and nodes that are initially arbitrarily set, then used the model with the highest accuracy among them. It consists of a total of six Hidden Layers, each of which is designed as Input: 592, Hidden Layer 1:900, Hidden Layer 2:1200, Hidden Layer 3:600, Hidden Layer 4:200, Hidden Layer 5:50, Hidden Layer 6:10, Output: 4. We show the structure of the proposed neural network in Figure 11.

Figure 11. Proposed Neural Network Structure.

4. Experiment

4.1. Experiment Environment

The experimental environment used two CPU Intel i7-8550, 32GB RAM, and 2080ti GPU RTX, followed by Window10, Matlab 2020a, and the library for learning was Caffe2. In the database, a total of 11,620 images were used for training images labeled with four materials, including concrete cement: 4739, health complete: 2643, irregular metal: 2500 images, of which 22 images were used randomly. Learning has verified the structure of the proposed algorithm and the performance of the algorithm compared to GoogleNet.

4.2. Detection of Roof Areas

Roads should be detected in satellite images for roof detection. Figure 12 presents the original image used for detection.

Figure 12. Satellite images for roof detection.

To detect areas with roofs in satellite images, lines consisting of straight lines and curves are detected. Figure 13a shows the lines detected in the image while Figure 13b depicts the roofed area using the lines observed.

(a)

(b)

Figure 13. Detect building areas (a) detect roads in images; (b) detect areas where buildings are located.

If the area of the building is detected, the building must be located in the detected area. If you have three-dimensional data at this time, you can detect it using height. If there are no 3D data, it can be detected by using noise and roof patterns. Figure 10 illustrates the region where structures are most likely to be found. The large-valued part of Figure 14 is roof-patterned, roof-colored, and is expected to have buildings in the area because of different noise from the surrounding environment.

Figure 14. Area division using noise in the image.

Figure 15 depicts the part of the image where the value varies due to gradients. To detect a change in value, add gradient operations to each X and Y axis. The region where the building is located is divided by each building using this boundary point. To detect the buildings, detect the edges of the separated structures.

Figure 15. Area division using noise in the image.

Figure 16 reveals the discovery of all buildings in the satellite image of Figure 12. We store both the location and the area where the building is located in the image and use this image data to classify the material of the roof with the CNN algorithm.

Figure 16. Building detected by satellite images.

4.3. Roof Material Classification

The CNN algorithm is applied to the representation of the building observed in Section 4.2. Figure 17 exposes an image of a single building from a satellite image. The green-roofed building extracted from the red circle section is applied to the convolutional layer to find the feature point.

Figure 17. The roof of a building separated from the image.

Figure 18 demonstrates a featured image of the roof of the extracted building detected in the convolutional layer. The material of the building is classified using these detected featured images. Features vary depending on the building's material, pattern, and noise, allowing the roof to be classified.

Figure 18. Detected Roof (a) Roof Original Image, (b) Features Detected by the Convolution Layer.

These detected roof data are used as learning data or as data for detection.

4.4. Learning for Roof Detection

The database used for learning is used as in Section 4.1. We learn the proposed CNN structure and GoogleNet for roof detection using the same database and compare the results.

For learning, the fully connected parameter was set as shown in Table 4, and the proposed CNN model and GoogleNet used the same fully connected. The proposed parameters include a total of six hidden layers.

Table 4. Hyperparameter applied to training.

Hidden Layer	6
Hidden Layer 1 Node	900
Hidden Layer 2 Node	1200
Hidden Layer 3 Node	600
Hidden Layer 4 Node	200
Hidden Layer 5 Node	50
Hidden Layer 6 Node	10
Dropout	50%
Weight Initialization	He Method
Active Function	Relu
Batch Size	8
Number of Epochs	70

Figure 19 shows the proposed CNN model and training, testing accuracy using GoogleNet. Batch Size is set to 8 and Epochs to 70. Since the validation data are randomly selected from the training data, we iterate in this way a total of 20 times to verify that the training data are well trained even when changed. Figure 19 shows higher accuracy of the proposed CNN model than GoogleNet in the proposed parameters, and Table 5 shows the accuracy results of 20 iterations of learning. Accuracy varies by as little as 5% and as much as 7%.

Figure 19. Roof Learning Proposed CNN, GoogleNet Accuracy Results.**Table 5.** Learning Training, Validation, Testing Accuracy result.

	Training Accuracy	Validation Accuracy	Testing Accuracy
Proposed CNN	97.4% $\pm$ 0.54%	78.7% $\pm$ 0.95%	73.3% $\pm$ 1.002%
GoogleNet	92.5% $\pm$ 0.72%	71.3% $\pm$ 0.84%	68.6% $\pm$ 1.16%

Table 6 shows the accuracy of CNN and GoogleNet, proposed by learning material. Material-specific accuracy shows that CNNs, which propose training, validation, and testing, show high accuracy of 3–9%, making it suitable for the CNN model proposed by the corresponding structure and parameters.

Table 6. Learning Training, Validation, Testing Accuracy result by material.

		Concrete Cement	Healthy Metal	Incomplete	Irregular Metal
Proposed CNN	Training Accuracy	98.7%	97.5%	96.7%	94.1%
	Validation Accuracy	83.4%	80.3%	77.2%	76.0%
	Testing Accuracy	77.8%	74.6%	72.1%	70.3%
GoogleNet	Training Accuracy	95.6%	94.2%	91.1%	90.6%
	Validation Accuracy	74.5%	73.4%	70.6%	68.3%
	Testing Accuracy	70.3%	70.1%	66.7%	65.1%

We select precision, recall, and F1 to quantitatively evaluate the performance of the model. Table 7 shows the resulting values for each performance metric. Precision indicators showed good performance over 95%, while recall also showed 87%. F1-score also shows over 91% performance, showing that it is a fully practical model.

Table 7. CNN model Performance Indicators.

	Precision	Recall	F1 Score
Proposed CNN	0.97	0.87	0.91
GoogleNet	0.96	0.81	0.87

Figure 20 shows test accuracy by batch size. The smaller the batch size, the higher the accuracy of the learning, but the longer the learning takes, the more appropriate batch size needs to be set. In the proposed CNN structure, the batch size is set to 8. Batch size is set to 8 because there is no significant difference from the result of having a batch size smaller than 8 on the graph, but the error becomes larger than the existing one.

Figure 20. Learning results with different batch sizes.

Table 8 shows the accuracy of the number of epochs. With fewer epochs set, termination is likely before learning is completed, leading to underfitting, and with too many epochs, there is likely to be overfitting. Therefore, proper epoch settings are needed. Table 8 shows the learning accuracy and error repeated 20 times for each epoch to determine how accurate the learning is and how it is learned by each epoch. The error range of accuracy is reduced to 70 with certainty, and then the error range converges with accuracy as soon as 70 is exceeded, indicating that the learning is complete if epochs is 70 or higher. In this paper, we show that setting epochs to 70 is appropriate because epochs become overfitting even if they are too large.

Table 8. Learning Test Accuracy by epoch.

	Epochs: 50	Epochs: 60	Epochs: 70	Epochs: 80	Epochs: 90
Proposed CNN	71.8% $\pm$ 1.61%	73.1% $\pm$ 1.2%	73.3% $\pm$ 1.002%	73.5% $\pm$ 0.78%	73.3% $\pm$ 0.77%
GoogleNet	67.6% $\pm$ 1.66%	68.5% $\pm$ 1.25%	68.6% $\pm$ 1.16%	68.3% $\pm$ 0.94%	68.5% $\pm$ 0.95%

5. Conclusions

The roof of a building can be used as a measure to determine whether the building is complete or incomplete. If there is a problem with the building, major casualties such as disasters or collapse of the building can occur, so analysis of the material used for the roof of the building can be a measure of the building situated in the area.

In order to obtain images of the roofs of buildings in large areas, satellite images must be used to extract and use the necessary information because they are very large in terms of resolution and size and contain a wide range of information. Information on the roof should be extracted from satellite images. Buildings can be easily distinguished if the original image contains three-dimensional information or if the height can be known using depth map, but in the absence of such information, the building must be detected first.

The conditions of the area in which the building is located were determined. Since the color information is distinct or high from the natural atmosphere due to the roof, it is separated into roads, etc., and has different noise elements from the surrounding area. These conditions are used to detect areas with buildings from satellite images and to extract images of roofs.

We classify the material of the roof using the results learned with the proposed CNN structure using the image of the roof. The roof material was divided into two categories: widely used materials and incomplete buildings. Comparing the results learned with the proposed CNN model with the existing structure of GoogleNet, training, validation, and testing accuracy showed a 5–7% performance improvement. When we look at the proposed structure precision, recall, and F1-score, better performance metrics than conventional models are observed, and the learned material-specific accuracy also has a 9% accuracy improvement. Via the material of the roof, it is possible to investigate and collect intelligence in a wide range of areas, such as emergency preparedness or construction complement, using such confidential information from satellite photographs. In the future, we aim to investigate algorithms that can be derived from satellite images, such as surrounding environment road signs and building details, and to evaluate conditions more accurately using algorithms that learn them.

Author Contributions: J.K. and H.B. conceived and designed the paper concept. J.K. made the formal analysis. H.B. and H.K. created the methodology. J.K. and H.K. wrote the paper. H.B. provided the resource and software. S.G.L. supervised and edited the manuscript for submission. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rajkumar, S.; Malathi, G. A comparative analysis on image quality assessment for real time satellite images. *Indian J. Sci. Technol.* **2016**, *9*, 34. [\[CrossRef\]](#)
2. Ayhan, B.; Kwan, C. Tree, Shrub, and Grass Classification Using Only RGB Images. *Remote Sens.* **2020**, *12*, 1333. [\[CrossRef\]](#)
3. Ayhan, B.; Kwan, C.; Larkin, J.; Kwan, L.; Skarlatos, D.; Vlachos, M. Deep learning model for accurate vegetation classification using RGB image only. *Geospatial Informatics X. Int. Soc. Optics Photonics* **2020**, *11398*, 113980H.
4. Zhao, K.; Kang, J.; Jung, J.; Sohn, G. Building extraction from satellite images using mask R-CNN with building boundary regularization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 247–251.
5. Dymkova, S.S. Conjunction and synchronization methods of earth satellite images with local cartographic data. In Proceedings of the 2020 Systems of Signals Generating and Processing in the Field of on Board Communications, Moscow, Russia, 19–20 March 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–7.
6. AlMarzooqi, M.; AlNaqbi, A.; AlMheiri, A.; Bezawada, S.; Mohamed, E.A.; Zaki, N. Increase the Exploitation of Mars Satellite Images Via Deep Learning Techniques. In Proceedings of the 2018 International Conference on Robotics, Control and Automation Engineering, Beijing, China, 26–28 December 2018; pp. 171–175.
7. Torres-Sánchez, J.; López-Granados, F.; Borra-Serrano, I.; Peña, J.M. Assessing UAV-collected image overlap influence on computation time and digital surface model accuracy in olive orchards. *Precis. Agric.* **2018**, *19*, 115–133. [\[CrossRef\]](#)
8. Czyńska, K. High Precision Visibility and Dominance Analysis of Tall Building in Cityscape-on a Basis of Digital Surface Model. In Proceedings of the 36th eCAADe Conference, Lodz, Poland, 17–21 September 2018; pp. 481–488.
9. Alganci, U.; Besol, B.; Sertel, E. Accuracy assessment of different digital surface models. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 114. [\[CrossRef\]](#)
10. Yan, Y.; Gao, F.; Deng, S.; Su, N. A hierarchical building segmentation in digital surface models for 3D reconstruction. *Sensors* **2017**, *17*, 222. [\[CrossRef\]](#)
11. Widyaningrum, E.; Lindenbergh, R.C.; Gorte, B.G.H.; Zhou, K. Extraction of building roof edges from LiDAR data o optimize the digital surface model for true orthophoto generation. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2018**, *42*. [\[CrossRef\]](#)
12. He, X.; Wang, A.; Ghamisi, P.; Li, G.; Chen, Y. LiDAR data classification using spatial transformation and CNN. *IEEE Geosci. Remote Sens. Lett.* **2018**, *16*, 125–129. [\[CrossRef\]](#)
13. Xia, J.; Yokoya, N.; Iwasaki, A. Fusion of hyperspectral and LiDAR data with a novel ensemble classifier. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 957–961. [\[CrossRef\]](#)
14. Wei, Y.; Ding, Z.; Huang, H.; Yan, C.; Huang, J.; Leng, J. A non-contact measurement method of ship block using image-based 3D reconstruction technology. *Ocean. Eng.* **2019**, *178*, 463–475. [\[CrossRef\]](#)
15. Xu, Y.; John, V.; Mita, S.; Tehrani, H.; Ishimaru, K.; Nishino, S. 3D point cloud map based vehicle localization using stereo camera. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 487–492.
16. Wang, H.; Zhou, M.X.; Zheng, W.Z.; Shi, Z.B.; Li, H.W. 3D machining allowance analysis method for the large thin-walled aerospace component. *Int. J. Precis. Eng. Manuf.* **2017**, *18*, 399–406. [\[CrossRef\]](#)
17. Liu, Y.; Wang, C.; Song, Z.; Wang, M. Efficient global point cloud registration by matching rotation invariant features through translation search. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 448–463.
18. Muresan, O.; Pop, F.; Gorgan, D.; Cristea, V. Satellite image processing applications in MedioGRID. In Proceedings of the 2006 Fifth International Symposium on Parallel and Distributed Computing, Timisoara, Romania, 6–9 July 2006; IEEE: Piscataway, NJ, USA, 2006; pp. 253–262.
19. Gorgan, D.; Bacu, V.; Stefanut, T.; Rodila, D.; Mihon, D. Earth Observation application development based on the Grid oriented ESIP satellite image processing platform. *Comput. Stand. Interfaces* **2012**, *34*, 541–548. [\[CrossRef\]](#)
20. Kussul, N.; Shelestov, A.; Skakun, S. Grid system for flood extent extraction from satellite images. *Earth Sci. Inform.* **2008**, *1*, 105. [\[CrossRef\]](#)
21. Chang, N.B.; Bai, K.; Chen, C.F. Smart information reconstruction via time-space-spectrum continuum for cloud removal in satellite images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 1898–1912. [\[CrossRef\]](#)
22. Durand, S.; Malgouyres, F.; Rougé, B. Image deblurring, spectrum interpolation and application to satellite imaging. *ESAIM Control Optim. Calc. Var.* **2000**, *5*, 445–475. [\[CrossRef\]](#)
23. Jianwen, M.; Xiaowen, L.; Xue, C.; Chun, F. Target adjacency effect estimation using ground spectrum measurement and Landsat-5 satellite data. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 729–735. [\[CrossRef\]](#)
24. Sellami, A.; Farah, I.R. Spectra-spatial Graph-based Deep Restricted Boltzmann Networks for Hyperspectral Image Classification. In Proceedings of the 2019 Photonics & Electromagnetics Research Symposium-Spring (PIERS-Spring), Rome, Italy, 17–20 June 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1055–1062.
25. Choi, J.; Park, H.; Kim, D.; Choi, S. Unsupervised change detection of KOMPSAT-3 satellite imagery based on cross-sharpened images by Guided filter. *Korean J. Remote Sens.* **2018**, *34*, 777–786.
26. Oh, J.; Lee, C. Epipolar Resampling Module for CAS500 Satellites 3D Stereo Data Processing. *Korean J. Remote Sens.* **2020**, *36*, 939–948.

27. Yuan, B.; Han, L.; Gu, X.; Yan, H. Multi-deep features fusion for high-resolution remote sensing image scene classification. *Neural Comput. Appl.* **2021**, *33*, 2047–2063. [[CrossRef](#)]
28. Kashani, A.G.; Graettinger, A.J. Cluster-based roof covering damage detection in ground-based lidar data. *Autom. Constr.* **2015**, *58*, 19–27. [[CrossRef](#)]
29. He, M.; Zhu, Q.; Du, Z.; Hu, H.; Ding, Y.; Chen, M. A 3D shape descriptor based on contour clusters for damaged roof detection using airborne LiDAR point clouds. *Remote Sens.* **2016**, *8*, 189. [[CrossRef](#)]
30. Sampath, A.; Shan, J. Building roof segmentation and reconstruction from LiDAR point clouds using clustering techniques. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2008**, *37*, 279–284.
31. Taherzadeh, E.; Shafri, H.Z. Development of a generic model for the detection of roof materials based on an object-based approach using WorldView-2 satellite imagery. *Adv. Remote Sens.* **2013**, *2013*. [[CrossRef](#)]
32. Liu, Z.J.; Wang, J.; Liu, W.P. Building extraction from high resolution imagery based on multi-scale object oriented classification and probabilistic Hough transform. In Proceedings of the 2005 IEEE International Geoscience and Remote Sensing Symposium, IGARSS'05, Seoul, Korea, 25–29 July 2005; IEEE: Piscataway, NJ, USA, 2005; pp. 2250–2253.
33. Beaudoin, N.; Beauchemin, S.S. An accurate discrete Fourier transform for image processing. In *Object Recognition Supported by User Interaction for Service Robots*; IEEE: Piscataway, NJ, USA, 2002; pp. 935–939.