TELKOMNIKA, Vol.16, No.6, December 2018, pp.3034~3044
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v16i6.10157

Inclined Image Recognition for Aerial Mapping using Deep Learning and Tree based Models

Muhammad Attamimi*, Ronny Mardiyanto, Astria Nur Irfansyah
Department of Electrical Engineering,
Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
*Corresponding author, e-mail: attamimi@ee.its.ac.id

Abstract

One of the important capabilities of an unmanned aerial vehicle (UAV) is aerial mapping. Aerial mapping is an image registration problem, i.e., the problem of transforming different sets of images into one coordinate system. In image registration, the quality of the output is strongly influenced by the quality of input (i.e., images captured by the UAV). Therefore, selecting the quality of input images becomes important and one of the challenging task in aerial mapping because the ground truth in the mapping process is not given before the UAV flies. Typically, UAV takes images in sequence irrespective of its flight orientation and roll angle. These may result in the acquisition of bad quality images, possibly compromising the quality of mapping results, and increasing the computational cost of a registration process. To address these issues, we need a recognition system that is able to recognize images that are not suitable for the registration process. In this paper, we define these unsuitable images as "inclined images," i.e., images captured by UAV that are not perpendicular to the ground. Although we can calculate the inclination angle using a gyroscope attached to the UAV, our interest here is to recognize these inclined images without the use of additional sensors in order to mimic how humans perform this task visually. To realize that, we utilize a deep learning method with the combination of tree-based models to build an inclined image recognition system. We have validated the proposed system with the images captured by the UAV. We collected 192 images and labelled them with two different levels of classes (i.e., coarse- and fine-classification). We compared this with several models and the results showed that our proposed system yielded an improvement of accuracy rate up to 3%.

Keywords*: aerial mapping, deep learning, image classification, image registration, tree-based models*

1. Introduction

In the era of information and communication technology (ICT), almost all research fields have been growing rapidly. Significant impact has been experienced in the field of robotics including Unmanned Aerial Vehicle (UAV). There is an extraordinarily rapid research and development on UAV technology due to its large possibilities that can be explored and exploited. One important capabilities of UAV is to produce a map of an area through aerial photogrammetry, or commonly referred as aerial mapping.

Aerial mapping by UAV has a large number of challenges and issues to address. Particularly, aerial mapping by utilizing cameras is supported by visual sensing technologies. Generally, aerial mapping is an image registration problem. Image registration problem is the problem of transforming different sets of images into one coordinate system [1]. The quality of the output of the registration system is greatly affected by the quality of the images inputted to the system (i.e., images captured by the UAV). Therefore, selecting the quality of the input images becomes an important task in aerial mapping. Moreover, it also becomes one of the challenging tasks in the field because sorting out the images to be processed efficiently is not trivial considering groundtruth in aerial mapping is not given before the UAV flies to take the images. On the other hand, generally, the UAV will fly and take images in sequence regardless of its flight orientation and roll angle, and therefore the suitability and quality of the acquired images will vary. This will result in: 1) the quality of mapping results may become bad, and 2) the computational cost of registration process becomes high. These problems form the background of our study of the quality of the images captured by the UAV during flight. Our approach is based on an image classification strategy.

To address the issues mentioned above, a recognition system is needed. In this paper, we define such image as "inclined images," i.e., images captured by the UAV which are not

Received June 22, 2018; Revised September 15, 2018; Accepted October 29, 2018

perpendicular to the ground. On the contrary, images that are perpendicular to the ground are called “normal images,” which are the target input of aerial mapping. An example of a “normal image” and an “inclined image” is respectively shown in lefttop and righttop of Figure 1. One can see from the figure that we can also classify the “inclined image” with some fine definitions such as, “low-inclined image”, “medium-inclined image,” and “high-inclined image” (see the right bottom of Figure 1). Our objective is to give the UAV more detailed information about the “inclined image.” This motivates us to build a tree-based model shown in Figure 1. We can see that from the “root,” there are two nodes that are a “normal image” and a “inclined image” which is defined as coarse model; and each of the parent’s node has respectively one and three child’s nodes. These lower nodes belong to the fine model.

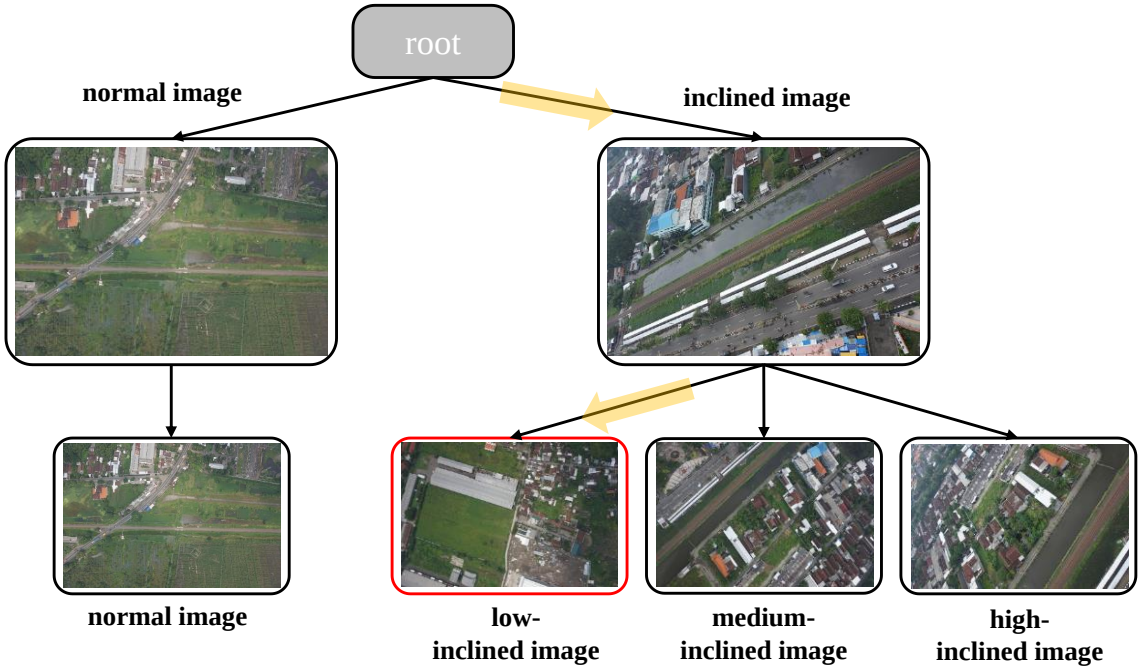

Figure 1. Images captured by the UAV is classified hierarchically based on a tree structure. Our objective is to recognize “inclined image,” and is to give a fine guest such as “low inclined image” (depicted in the red frame).

In this paper, although we can calculate the inclination angle using a gyroscope attached to the UAV, our interest here is to recognize the images exclusively based on the acquired images without the use of such sensor, similar to how humans do this task, i.e., people can estimate somehow that images in the rightside of Figure 1 are inclined. Here, we utilize a deep learning method because it is the state-of-the-art of image classification task. To realize our proposed inclined image recognition, first we collect the data captured by UAV and label them. It should be noted that labels are given according to the tree-basedtree-based models shown in Figure 1. Then, we perform a transfer learning on the collected data to build the recognition system consisting of two-leveled models (i.e., coarse model and fine model), that should be able to distinguish the image whether it is the normal one or not. The work in [2] inspired us to build a hierarchical image recognition system. We remedy the results in the lower nodes considering the hierarchical structure of the tree-basedtree-based models by a simple Bayesian approach. The final output of the system is shown in Figure 1 with the red frame.

There have been numerous studies related to general image classification in the literature [2–10]. However, studies involving UAV are not so common. Existing literature are more directed to a specific image classification and not for improvement of image registration for mapping region [11]. Therefore, in our work, we propose an image recognition system that can be utilized to improve the image registration process on the UAV. The expected benefit of this

study is to produce an image registration system which is better and faster, and can be implemented in UAV for efficient aerial mapping.

2. Related Works

The focus of this study is the introduction of inclined images captured by UAV in aerial mapping. Most of the existing research on aerial mapping is related to photogrammetry, particularly air photogrammetry [12–14]. In the scope of UAV, more emphasis had been on camera calibration [13] and also global mapping using various types of sensors [14]. However, existing work reporting aerial photogrammetry utilizing image recognition is still relatively rare.

In image recognition, we select the method of image classification that can categorize target images into several categories or better known as visual category recognition. Category recognition has also been done for quite a while involving several generations; from the simple technique by utilizing multiple filters, Neural Network (NN), up to feature-based methods which can represent objects globally as in [3] or locally as [4, 5]. In particular, the usage of local features, e.g. the presence of Bag-of-Keypoints (BoK) [6] paved the way greatly for category recognition. Technology continues to have improved well into the era of big data; where the massive amount of data is utilized for a particular job. In this era, NN which had receded several decades is now experiencing a revival with a framework involving a new learning algorithm which is known as deep learning. For visual recognition, Convolutional Neural Networks (CNN) has become state-of-the-art in the field [7].

CNN has been ranked top in the category of visual recognition for household objects, the classification of flora and fauna, and other natural photographs [15]. However, the utilization of CNN in the field of UAV for image classification problem has still not been widely reported. Therefore, in this work we take the advantage of CNN's superiority within the deep learning framework to solve the problem of image classification. The results of this classification is then used for image registration in the aerial mapping by an UAV.

The remainder of this paper is organized as follows. An overview of the proposed method followed by the details of each process needed to realize the inclined image recognition system is presented in section 3. The experimental setting and results are discussed in section 4. Finally, section 5 concludes this study.

3. Proposed Method

In this study, we model the problem of finding the inclined images captured by an UAV as a model which is built in the deep learning framework through a transfer learning with the combination of tree-based models. The models used in this paper is illustrated in Figure 2. There are three main process that are, 1) a transfer learning scenario, 2) classification of each level (which are coarse classes and fine classes) that are based on a deep learning framework, and 3) a tree-based models to infer the final results. Coarse classes consist of “normal image” class and “inclined image” class, whereas fine classes consist of “normal image” class, “low-inclined image” class, “medium-inclined image” class, and “high-inclined image” class. Using the proposed method, the wrong classification can be corrected (see the red bars in fine class model). tree-based

To perform a transfer learning, the dataset that consists of images captured by an UAV is needed. Then, the dataset, is divided into a training dataset and the validation dataset. In this study, as a comparation we used two models of deep learning (i.e., *AlexNet* and *GoogLeNet*) which is well known to have good performance in the tasks of image classification. The transfer learning is done for each level of classification, i.e., a coarse classification which consist of two classes (i.e., “normal image” and “inclined image”) and a fine classification which consist of four classes (i.e., “normal image”, “low-inclined image”, “medium-inclined image”, and “high-inclined image”).

After the transfer learning is completed, the models can be used to estimate the class of an input image. Thanks to the model, we can get the probability or class estimation score. These probabilities can then be used as an input to the tree-based models.

The final step of the proposed inclined image recognition is combining the results of each level through a simple Bayesian approach. We can see from Figure 2 that using the proposed method the incorrect result of fine class can be corrected (see red bars in Figure 2).

Our proposed method is especially effective when the classification score is ambiguous. The detailed process of those phases is described as follows.

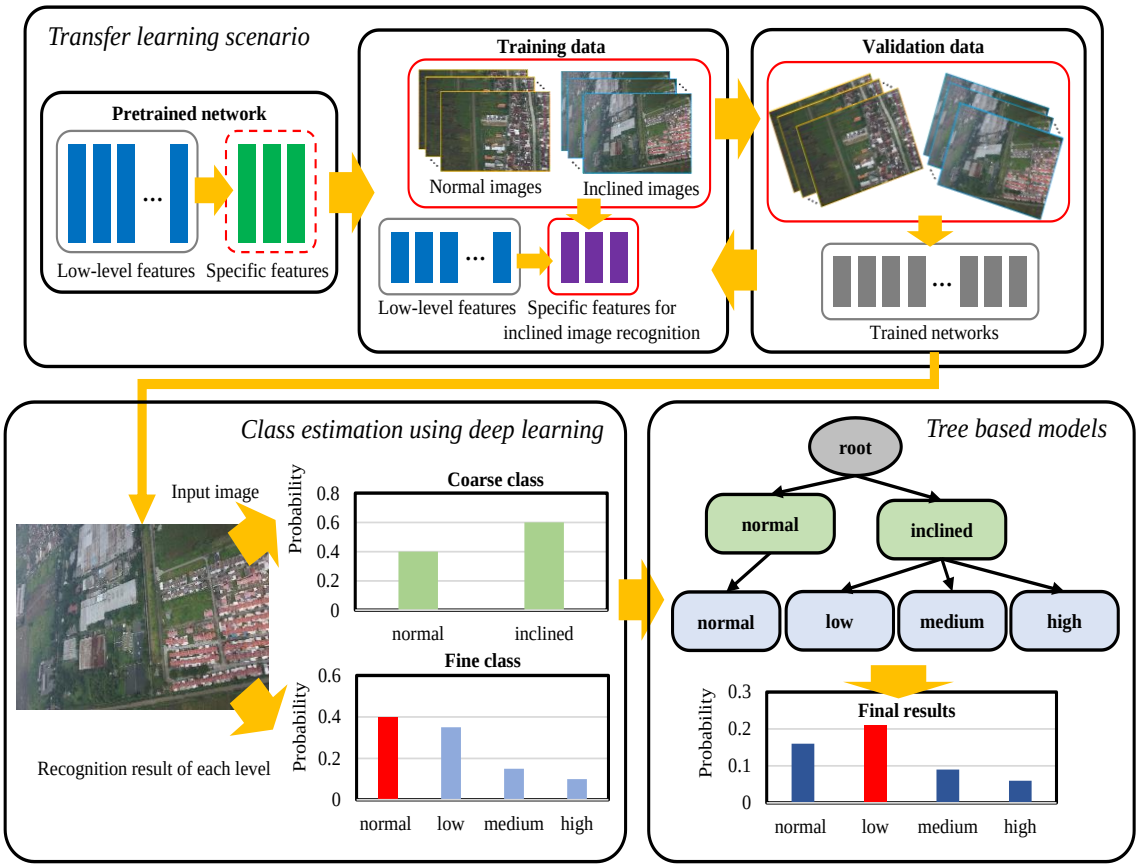

Figure 2. Overview of proposed inclined image recognition system.

3.1. Transfer Learning

Transfer learning is a method where a model developed for a task is reused as the initial points or parameters for a model on another task. In machine learning, especially in a deep learning framework, it is common to use a pretrained network as the initial parameters for training a desired network (i.e., a network for the incline image recognition task). In this paper, we use and compare *AlexNet* and *GoogLeNet* as pretrained networks due to their capability in visual recognition tasks.

Both of the pretrained networks, *AlexNet* and *GoogLeNet,* are networks that has been trained on over a million images and can classify images into 1000 object categories (such as keyboard, coffee mug, pencil, and many animals). The network has learned rich feature representations for a wide range of images. Therefore, the low-level features showed in Figure 2 can be utilized to build a good network. The network takes an image as input and outputs a label for the object in the image together with the probabilities for each of the object categories.

The *AlexNet* network used in this paper consists of 25 layers (see Table 1 for details information) including five convolutional layers and three fully connected layers. Almost in each convolutional layer, it comes along with a rectified linear unit (ReLU), a cross channel normalization, and a max pooling. The first layer is the image input layer, which is a vector with the size of size 227-by-227-by-3, where three is the number of color channels. The last three layers of the pretrained network are configured for 1000 classes, which is indicated a specific feature for a particular task. Therefore, these three layers must be fine-tuned for the new classification problem which are two classes (i.e., “normal image” class and “inclined image”

class) for coarse classification problem and four classes (i.e., “normal image” class, “low-inclined image” class, “medium-inclined image” class, and “high-inclined image” class) for fine classification problem. These layers are then transferred to the new classification task by replacing the last three layers with a fully connected layer, a softmax layer, and a classification output layer.

Table 1. The Pretrained Network AlexNet Used in this Paper.

Layer	Name	Operation	Details
1	'data'	Image Input	227x227x3 images with 'zerocenter' normalization
2	'conv1'	Convolution	96 11x11x3 convolutions with stride [4 4] and padding [0 0 0 0]
3	'relu1'	ReLU	ReLU
4	'norm1'	Cross Channel Normalization	cross channel normalization with 5 channels per element
5	'pool1'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
6	'conv2'	Convolution	256 5x5x48 convolutions with stride [1 1] and padding [2 2 2 2]
7	'relu2'	ReLU	ReLU
8	'norm2'	Cross Channel Normalization	cross channel normalization with 5 channels per element
9	'pool2'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
10	'conv3'	Convolution	384 3x3x256 convolutions with stride [1 1] and padding [1 1 1 1]
11	'relu3'	ReLU	ReLU
12	'conv4'	Convolution	384 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
13	'relu4'	ReLU	ReLU
14	'conv5'	Convolution	256 3x3x192 convolutions with stride [1 1] and padding [1 1 1 1]
15	'relu5'	ReLU	ReLU
16	'pool5'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0 0 0]
17	'fc6'	Fully Connected	4096 fully connected layer
18	'relu6'	ReLU	ReLU
19	'drop6'	Dropout	50% dropout
20	'fc7'	Fully Connected	4096 fully connected layer
21	'relu7'	ReLU	ReLU
22	'drop7'	Dropout	50% dropout
23	'fc8'	Fully Connected	1000 fully connected layer
24	'prob'	Softmax	softmax
25	'output'	Classification Output	crossentropyex with 'tench' and 999 other classes

The *GoogLeNet* used in this paper is a directed acyclic graph (DAG) which consists of 144 layers. DAG networks can have a more complex architecture where layers can have inputs from, or outputs to, multiple layers. Basically, the contents of each network/layer are almost the same with the ones in *AlexNet*, except there is also a depth concatenation process in the *GoogLeNet*.

3.2. Class Estimation using Deep Learning

Once the model is trained, we can use the model to estimate the input image. In this study, we deal with the classification problem. To cope with that, a softmax layer and then a classification layer must follow the final fully connected layer. The softmax function is the output unit activation function after the last fully connected layer for multi-class classification problem. Thanks to the softmax layer, the probability of each class can also be estimated. These probabilities can be used as estimation accuracy. If the accuracy is low, we can reconsider the results.

For a class estimation task, we can formulate the problems as follows. First, assume that $P(c|\theta, I)$ is the $probability$ of an input image I inferred as a class c for a trained networks θ (i,e., the output of softmax layer). Then, the classification result is the class with the highest probability c_{max}, which can be calculated as follows:

$$\mathrm{c}_{max} = argmax_c \, P(c|\theta, I). \quad (1)$$

In this study, we perform two levels (i.e. coarse class and fine class) of classification class with different number of classes. Both of them can be inferred using the Eq. (1). Moreover, the probability outputted from softmax layer can also be used as an input for tree-based models.

3.3. Tree based Models for Final Recognition Results

In this study, a tree-based model is used to reduce the error caused by the trained models. The idea here is to utilize the structure or the hieracy on tree-based models, i.e., the children nodes are constrained by their parent nodes. Here, we can formulate the problems as follows. It should be noted that the probabilities of each class in each level obtained in section 3.2 are used.

Assume that the probability of a level $l \in \{coarse, fine\}$ has a probability $P_l(c_l|\theta_l, I)$ for a class c_l. Here, we follow the Bayesian approach for independence probability which yield the final probability of fine class $P_{fine}(c_l|\theta, I)$ can be inferred as follows:

$$P_{fine}(c_l|\theta, I) = \prod_l P_l(c_l|\theta_l, I). \quad (2)$$

Then, the final class estimation of fine class c_{max}^{fine} is calculated as following.

$$c_{max}^{fine} = argmax_{c^{fine}} P_{fine}(c^{fine}|\theta, I)\,. \quad (3)$$

4. Experiments

To validate our proposed method, we have conducted several experiments. First, the experiment is to compare and report the result of transfer learning using the *AlexNet* model and *GoogLeNet* model. Second, we have also performed class estimation for each levels. Finally, we have implemented the tree-based models and tested the performance of our proposed inclined image recognition system. The details of experimental settings and the results are described as the following.

4.1. Experimental Settings

In this study, we used UAV Skywalker X8 Flying Wing [16]. The UAV is equipped with pixhawk to make it enable to fly autonomously. Our UAV is also able to fly up to 300 meters from the ground. The data used in this paper, was taken when the UAV flew at an altitude of 300 meters from the ground.

To capture the images, we used a smart phone's camera (i.e., SONY ILCE-5000) due to its simplicity. The camera has an exposure time 0.001 second, ISO-200, focal length 16 mm, and max aperture 3.6171875. The camera is able to acquire an image with a resolution of 5456 x 3632. We programmed the camera to capture the images at a speed of 1 fps. Then, we collected images to build the dataset. Here, we labelled 192 images with two different levels of classes, i.e., “normal image” class and “inclined image” class for coarse classification; and “normal image” class, “low-inclined image” class, “medium-inclined image” class, and “high-inclined image” class for fine classification. We divided the dataset into a training set which consists of 116 images and a test set which consists of 76 images. Figure 3 shows some examples of images in the dataset with the corresponding labels of each level classification task. The transfer learning of each levels was done using the training set.

The model used in this paper, is trained using a stochastic gradient descent (SGD) with momentum value of 0.9 and a mini-batch with size of 10 data. We have set the weight learning rate factor and bias learning rate factor of fully connected layer by 10 respectively. The initial learning rate is set to 0.0001. For comparison we conducted two different methods to set the data of mini-batch; i.e., 1) shuffle the training data before each training epoch, and shuffle the validation data before each network validation, this method is called as “every-epoch,” and 2) shuffle the training and validation data once before training, this method is called as “once.” We then fed the network for six epochs, and validated each process in transfer learning to get a better parameters.

4.2. Experimental Results

First, the training progress of each level of classification, i.e. coarse classification and fine classification shown in Figure 4. As we mentioned in section 4.1., we have used two pretrained networks, i.e., *AlexNet* and *GoogLeNet*, and we have also compared two mini-batch setting methods, i.e., “once” and “every-epoch.” From the figure we can see that the training started with low accuracy, the classification accuracy then gradually increased and ended with a

final value as the epoch has been reached. All of four cases mentioned in Figure 4 has the same tendency. In general, the classification results of coarse model were higher than the fine one due to the complexity of the problem.

Next, we have also tested the model of inclined image recognition with a test set. Figure 5 shows the confusion matrices of the classification results of each method. One can see that a coarse classification could be done better than the fine one because the problem was easy compared to the fine classification. We can also refer to Table 2 to see the accuracy of each model. The model (1) was the highest result in the coarse classification and the remaining methods were still achieved the accuracy about 90%. The classification errors were due to the similarness of those images that causes confusion to the model when deciding the correct classes. These results indicate that the first objective for building a coarse inclined image recognition can be achieved.

For a fine classification problem, we compared the model that uses the results of transfer learning with the proposed method that combines the results with a tree-based model. One can see from Table 2 that the classification results produced by all the models were boosted by the proposed method. The highest result was still the model (1).

Figure 6 shows the examples of input images which were estimated incorrectly by using a deep learning result only, but can be corrected by using the proposed method. Thanks to a Bayesian approach the probability of each class in the fine classification could be adjusted to the correct one. This adjustment is very effective especially for the results which were ambiguous. Overall, all of these results indicates that the proposed method performs well for an inclined image recognition task.

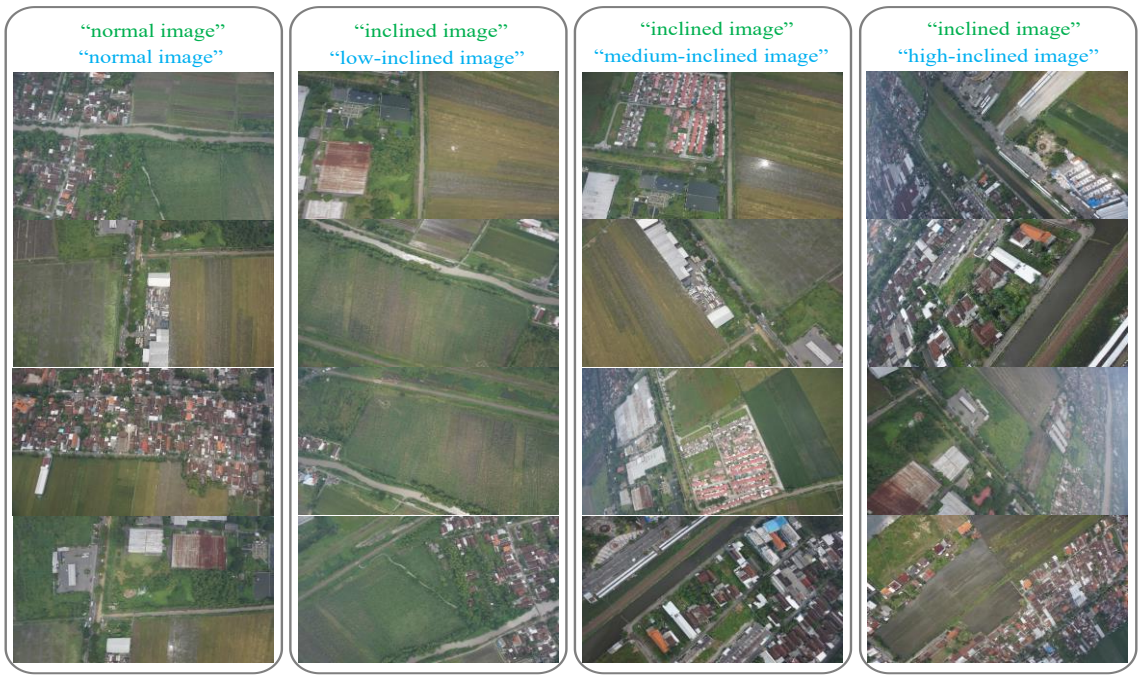

Figure 3. Examples of images in a training dataset. The texts written in green indicate coarse class (consists of two classes) whereas the blue ones indicate the fine class (consists of four classes).

Table 2. The Recognition Accuracies of the Models used in this Paper. Similar Color Indicates the Similar Classification Problems.

No.	Models	Classification problems		
		Coarse	Fine	Proposed
1.	AlexNet learned with "once" method	0.9342	0.82895	0.85526
2.	GoogLeNet learned with "once" method	0.8947	0.78947	0.80263
3.	AlexNet learned with "every-epoch" method	0.9079	0.80263	0.84211
4.	GoogLeNet learned with "every-epoch" method	0.8947	0.82895	0.84211

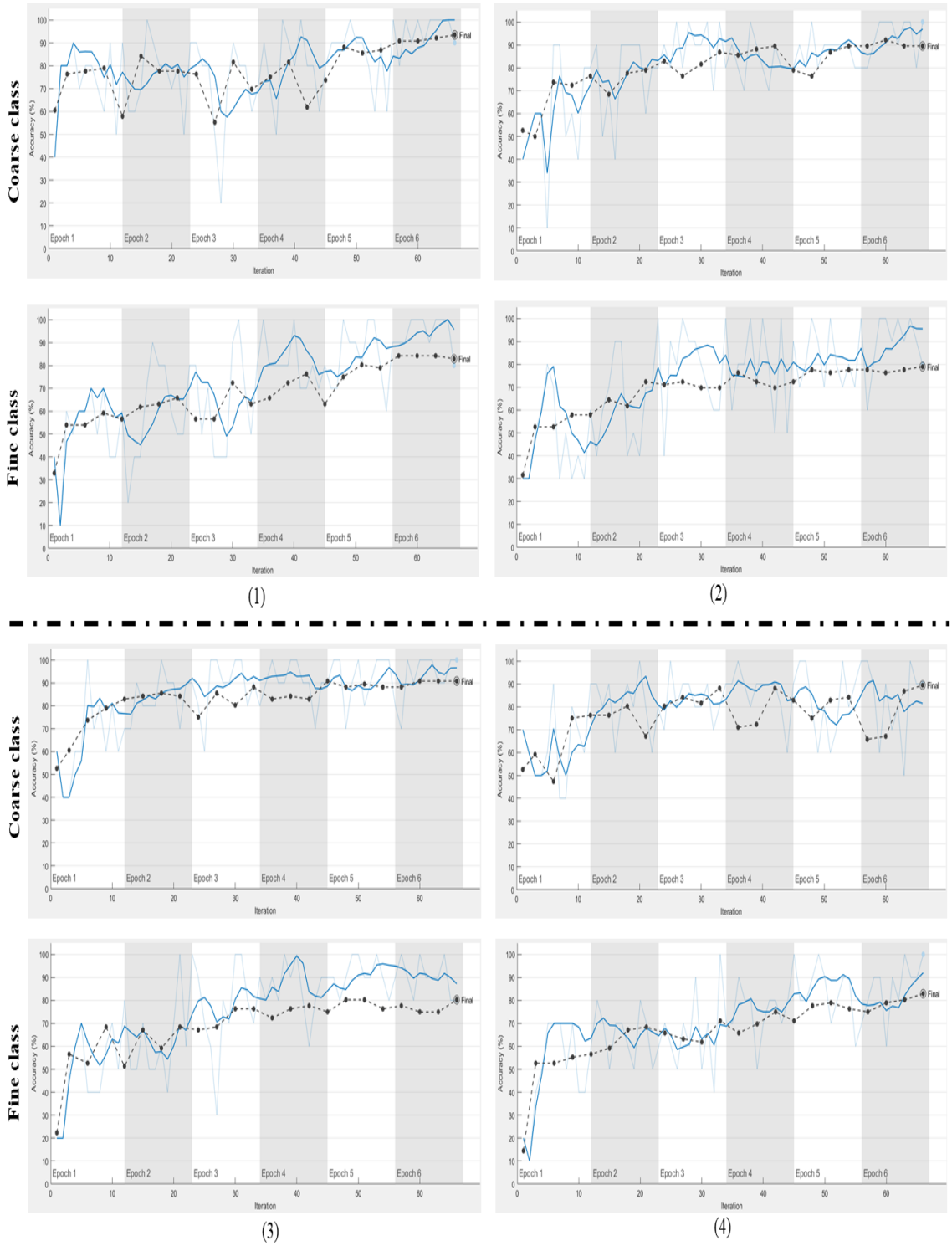

Figure 4. The training progress of transfer learning of each level of classification (i.e., coarse classification and fine classification). There are four learning set up, i.e., (1) learning using the *AlexNet* model with a "once" method, (2) learning using the *GoogLeNet* model with a "once" method, (3) learning using the *AlexNet* model with a "every-epoch" method, and (4) learning using the *GoogLeNet* model with a "every-epoch" method.

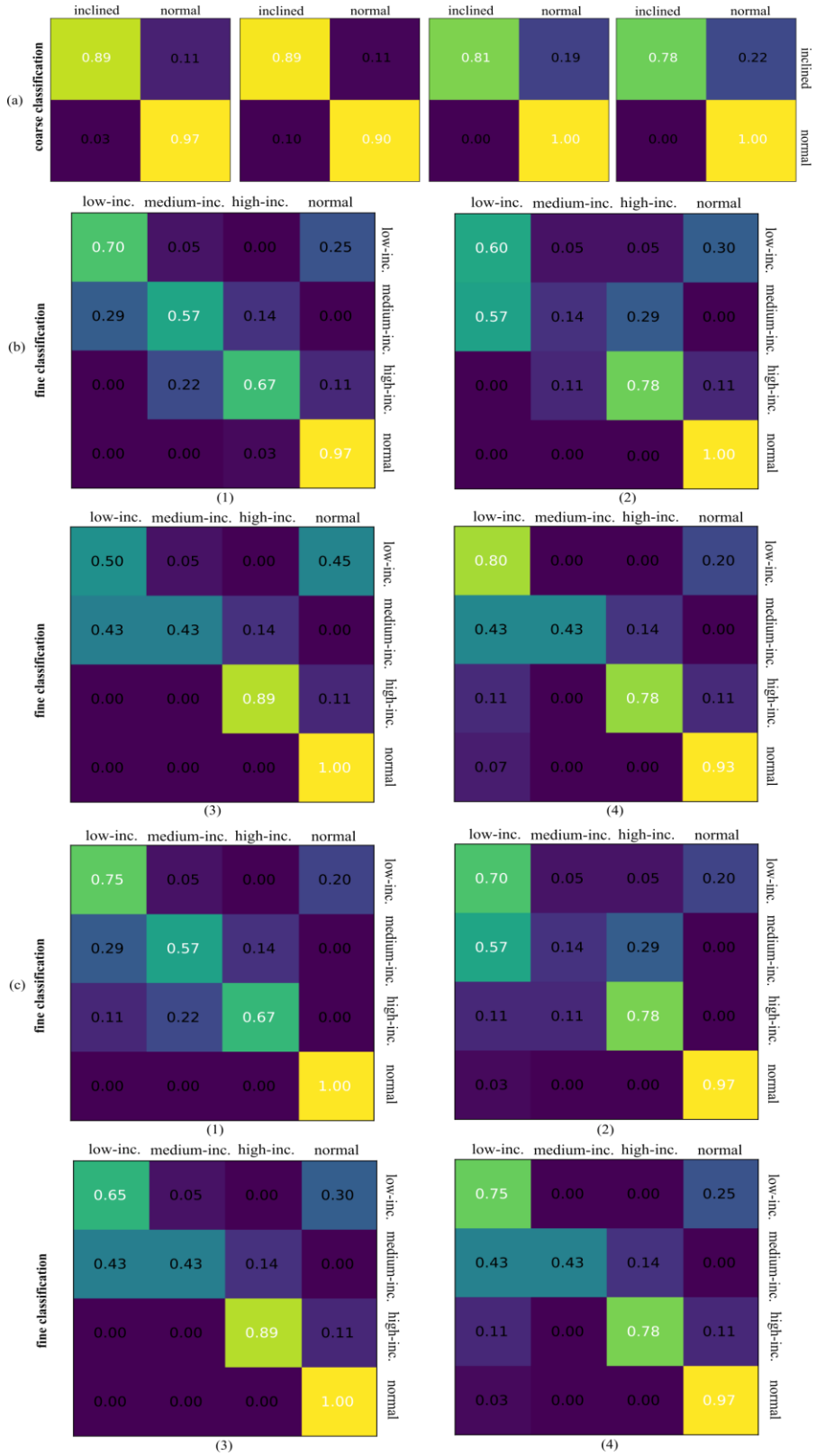

Figure 5. Confusion matrices of inclined image recognition tasks with the models: (1) *AlexNet* model learned with a "once" method, (2) *GoogLeNet* model learned with a "once" method, (3) *AlexNet* model learned with a "every-epoch" method, and (4) *GoogLeNet* model learned with a "every-epoch" method. The results of coarse classification is depicted in (a), whereas the fine classification is depicted in (b), the results of proposed method is depicted in (c).

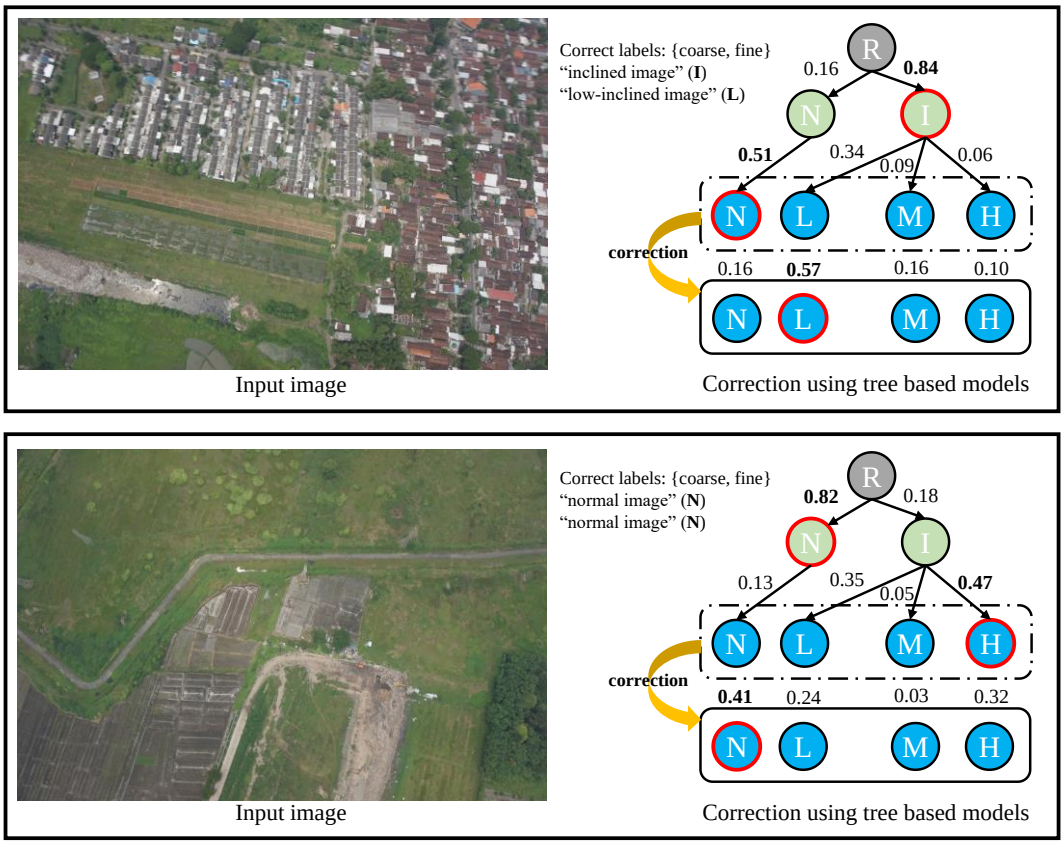

Figure 6. Examples of input images which were misclassified using a deep learning method only and were corrected by the proposed method that used tree-based models. The symbols "R", "N", "I", "L", "M", "H", denote root node, "normal image" class, "inclined image" class, "low-inclined image" class, "medium-inclined image" class, and "high-inclined image" class respectively. Correct labels indicate the corresponding correct labels of each classification problem (i.e., coarse classification and fine classification).

5. Conclusion

In this paper, we have introduced the problems in aerial mapping based on visual sensing. The problems were tackled by a proposed inclined image recognition system which is based on transfer learning utilizing a deep learning framework and a tree-based models which is based on a simple Bayesian approach. Our proposed method is able to recognize the inclined images in coarse way and the fine one. These abilities are important for the UAV to create a good map faster. We have collected the database using the UAV, and tested our proposed system. The results revealed that our proposed inclined image recognition system was able to perform recognition task with an accuracy of 93.42% for the coarse classification and could boost the recognition performance by 3% which yielded an accuracy of 85.52% for the fine classificaton results.

In the future, we are planning to develop a hierarchical inclined image recognition system which is able to estimate the inclination angle visually without the help of the sensor such as a gyroscope. Moreover, we also plan to develop a system to determine appropriate images for aerial mapping by adding the class in the recognition system such as adding the image that is captured in the foggy condition.

References

[1] Zitova B, Flusser J. Image registration methods: a survey. *Image and Vision Computing*. 2003; 21(11): 977–1000.

[2] Attamimi M, Nakamura T, Nagai T. *Hierarchical Multilevel Object Recognition Using Markov Model*. Proceedings of the 21st International Conference on Pattern Recognition, 2012.
[3] Osada R, Funkhouser T, Chazelle B, Dobkin D. Shape Distributions. *ACM Transactions on Graphics*. 2002; 21(4): 807–832.
[4] Lowe DG. Distinctive Image Features from Scale-Invariants Keypoints. *Journal of Computer Vision*. 2004;.60(2): 91–110.
[5] Vedaldi A, Fulkerson B. *VLFeat-An Open and Portable Library of Computer Vision Algorithms*. ACM Multimedia.
[6] Csurka G, Dance C, Fan L, Williamowski J, Bray C. *Visual Categorization with Bags of Keypoints*. Int. Workshop on Statistical Learning in Computer Vision. 2004: 1–22.
[7] Krizhevsky A. *ImageNet Classification with Deep Convolutional Neural Networks*. Proc. NIPS 2012.
[8] Lu D, Weng Q. A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*. 2007; 28(5): 823–870.
[9] Attamimi M, Araki T, Nakamura T, Nagai T. Visual Recognition System for Cleaning Tasks by Humanoid Robots. *International Journal of Advanced Robotic Systems: Humanoid*. 2013: 1–14.
[10] Abe K, Hieida C, Attamimi M, Nagai T, Shimotomai T, Omori T, Oka N. *Toward playmate robots that can play with children considering personality*. Proceedings of the second international conference on Human-agent interaction. 2014: 165–168.
[11] Szilvia K, Zoltan V, Bela M. Aerial Image Classification for the Mapping of Riparian Vegetation Habitats. *Acta Silv. Lign. Hung*. 2013; 9: 119–133.
[12] Abdullah T, Gillani SM, Qureshi HK, Haneef I. *Heritage Preservation using Aerial Imagery from light weight low cost Unmanned Aerial Vehicle (UAV)*. International Conference on Communication Technologies (ComTech). 2017: 201–205.
[13] Wu J, Zhou G, Li Q. *Calibration of Small and Low-Cost UAV Video System for Real-Time Planimetric Mapping*. IEEE International Symposium on Geoscience and Remote Sensing. 2006: 2068–2071.
[14] Liang Z, Zhang J, Xu X. *Design and Implement of Rotating Scanning Photograph Platform*. 2nd International Asia Conference on Informatics in Control, Automation and Robotics. 2010: 485–488.
[15] Cengil E, Çõnar A, Özbay E. *Image Classification with Caffe Deep Learning Framework*. International Conference on Computer Science and Engineering (UBMK). 2017.
[16] pixhawk.org, "Skywalker X8 Flying Wing," [Online]. Available: *https://pixhawk.org/platforms/planes/skywalker_x8*.