IMAGE LABELING FOR LIDAR INTENSITY IMAGE USING K-NN OF FEATURE OBTAINED BY CONVOLUTIONAL NEURAL NETWORK

Masaki Umemura†, Kazuhiro Hotta†, Hideki Nonaka‡ and Kazuo Oda‡

†Meijo University
1-501 Shiogamaguchi, Tempaku-ku, Nagoya 468-8502, Japan
E-mail: 120433015@ccalumini.meijo-u.ac.jp, kazuhotta@meijo-u.ac.jp

‡Asia Air Survey Co.,Ltd.
1-2-2 Manpukuji, Asao-ku, Kawasaki, Kanagawa, Jpan
E-mail: {hdk.nonaka, kzo.oda}@ajiko.co.jp

ICWG III/VII

KEYWORDS: Image labeling, Convolutional neural network, K-nearest neighbour and LIDAR intensity image

ABSTRACT:

We propose an image labeling method for LIDAR intensity image obtained by Mobile Mapping System (MMS) using K-Nearest Neighbor (KNN) of feature obtained by Convolutional Neural Network (CNN). Image labeling assigns labels (e.g., road, cross-walk and road shoulder) to semantic regions in an image. Since CNN is effective for various image recognition tasks, we try to use the feature of CNN (Caffenet) pre-trained by ImageNet. We use 4,096-dimensional feature at fc7 layer in the Caffenet as the descriptor of a region because the feature at fc7 layer has effective information for object classification. We extract the feature by the Caffenet from regions cropped from images. Since the similarity between features reflects the similarity of contents of regions, we can select top K similar regions cropped from training samples with a test region. Since regions in training images have manually-annotated ground truth labels, we vote the labels attached to top K similar regions to the test region. The class label with the maximum vote is assigned to each pixel in the test image. In experiments, we use 36 LIDAR intensity images with ground truth labels. We divide 36 images into training (28 images) and test sets (8 images). We use class average accuracy and pixel-wise accuracy as evaluation measures. Our method was able to assign the same label as human beings in 97.8% of the pixels in test LIDAR intensity images.

1. INTRODUCTION

It is important to properly make and update the Fundamental Geospatial Data for the maintenance of road (Hasegawa et al, 2013). A lot of attention has been paid to advanced driver assistance in nearest years. To realize it, we need to maintain the Fundamental Geospatial Data with high accurately and low cost.

Fundamental Geospatial Data of road is made by human now. However, manual process has some problems. Human cannot treat a large amount data, and there is the possibility of human error. In addition, since many people are required to make the Fundamental Geospatial Data of road, a lot of costs are required. Thus, automatic creation of the Fundamental Geospatial Data of road is required to reduce human burden and cost.

In this paper, we propose an automatic recognition method from LIDAR Intensity Image obtained by MMS (Novak, 1993). If this method is realized, we will convert the LIDAR intensity images into Fundamental Geospatial Data of road in the future. To create a map from LIDAR intensity images automatically, we need to recognize objects at each pixel in LIDAR intensity images. Thus, we propose an image labeling method which assigns label to each pixel in LIDAR intensity image using KNN (LeCun et al, 1998) of feature obtained by CNN (Dudani, 1976). We extract the feature by the Caffenet from semantic regions cropped from LIDAR intensity images, and we select top K similar regions cropped from training samples with a test region. Then, we vote the labels attached to top K similar regions to the test region.

In experiments, we use 36 LIDAR intensity images obtained by the MMS with ground truth labels. Those images include 9 categories. We use both class average accuracy and pixel-wise accuracy as evaluation measures. Our proposed method achieves 97.63% in class average accuracy and 74.96% in pixel-wise accuracy.

This paper is organized as follows. We explain the details of the proposed method using KNN of feature obtained by CNN in section 2. Evaluation results of our method are shown in section 3. Section 4 is for conclusions and future works.

2. PROPOSED METHOD

We automatically assign class labels to each pixel in LIDAR intensity images. Figure 1 shows the overview of our method. Since CNN is effective for various image recognition tasks, we try to use the feature obtained by CNN. We extract the feature obtained by CNN from semantic regions (e.g. 64x64 pixels) cropped from LIDAR intensity images. Since the similarity between features reflects the similarity of contents of regions, we can select top K similar regions cropped from training samples with a test region. Since regions in training images have manually-annotated ground truth labels, we vote the labels attached to top K similar regions to the test region. The class label with the maximum vote is assign to each pixel in the test image.

We explain feature extraction by CNN in section 2.1. Image labeling using KNN is explained in section 2.2.

This contribution has been peer-reviewed.
doi:10.5194/isprsarchives-XLI-B3-931-2016

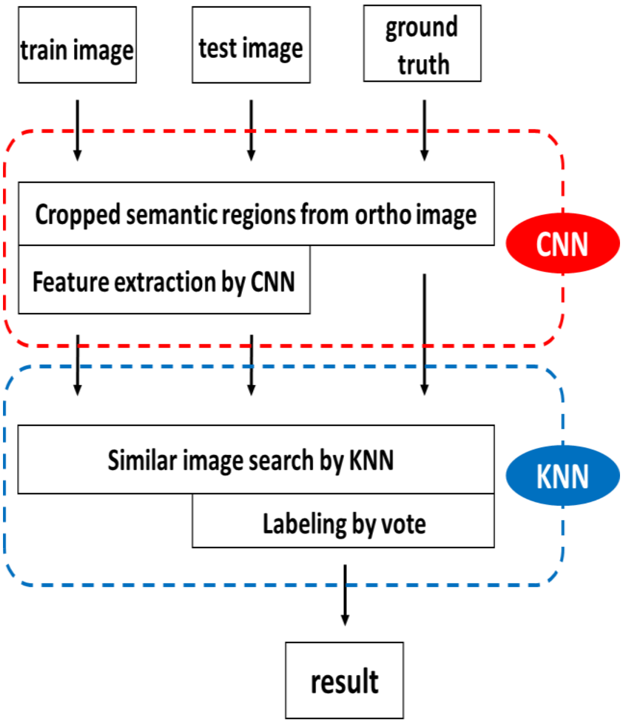
train image
test image
ground truth
Cropped semantic regions from ortho image
Feature extraction by CNN
CNN
Similar image search by KNN
Labeling by vote
KNN
result

Figure 1. The overview of our method.

2.1. Feature extraction by CNN

CNN is a kind of deep leaning method, and it is reported that high accuracy is obtained by CNN in various image recognition tasks (Krizhevsky et al, 2012). CNN consists of the convolution layers, pooling layers and fully connected layers. Convolution layers in CNN extract feature by filters obtained automatically through training. Pooling layers in CNN downsize the resolution to be robust to slight translation. Fully connected layers are the same as classical multi-layered perceptron (Jonathan et al, 2015).

In this paper, we use the Caffenet as CNN (Jia et al, 2014). Caffenet contains five convolution layers, three pooling layers and two fully connected layers. Caffenet was pre-trained by the ImageNet which contains 1,000 object categories. Caffenet gave top-1 accuracy 57.4% and top-5 accuracy 80.4% on the ImageNet. In this paper, we use the Caffenet as a feature extractor.

Since the Caffenet was optimized for the ImageNet (Deng et al, 2015), we optimize it for LIDAR intensity images by fine-tuning (Reyes et al, 2015). Since the number of neurons in the original Caffenet is 1000, we change the number of neurons at output layer to 9. In general, a large number of training samples is required to train CNN. Thus, we use 40,000 images for fine-tuning. If we crop the regions randomly, the number of road is large. Thus, we adjust the number of regions in each class to be nearly equal. The class label is put to each region manually. We confirmed that the classification accuracy improved by fine-tuning.

We use 4,096-dimensional feature at fc7 layer in the fine-tuned Caffenet as the descriptor of a region because the feature at fc7 layer has effective information for object classification. We extract the feature from all semantic regions cropped from LIDAR intensity images. Figure 2 shows the overview of feature extraction by CNN.

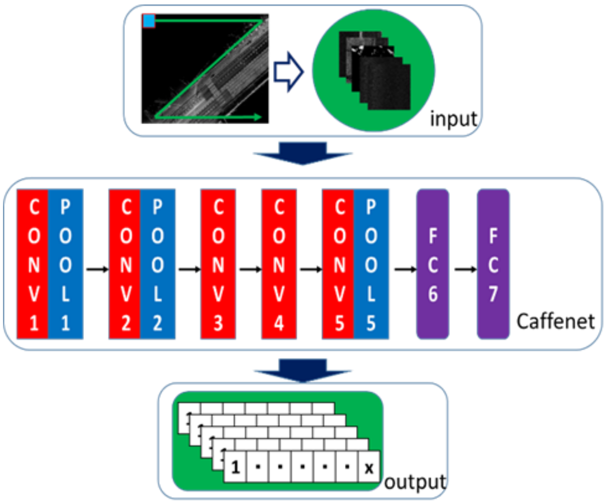

Figure 2. The overview of feature extraction by CNN.

2.2. Image labeling by KNN

Since the similarity between features reflects the similarity of contents of regions, we can select top K similar regions cropped from training samples with a test region by KNN. In addition, we know the distance of the regions with similar features. If we obtain K semantic regions from training samples, we also obtain the ground truth labels attached to the top K regions. Our method votes the ground truth labels of the K regions to the test region. In order to use the similarity for voting process, we use weighted vote according to the distance of region as

$$w_m = \frac{d_k - d_m}{d_k - d_1},$$

where d_m is the distance of the m-th nearest neighbour, d_1 is the distance of the most similar semantic region and d_k is the distance of the K-th nearest neighbor. Namely, the weight of the most similar region is 1 and the most unsimilar region is 0. We vote the weight to each pixel in test image. The voting process is performed with overlapped manner in the test image. The class label with the maximum vote is assigned to the pixel in the test images. Figure 3 shows the overview of Image labeling by KNN.

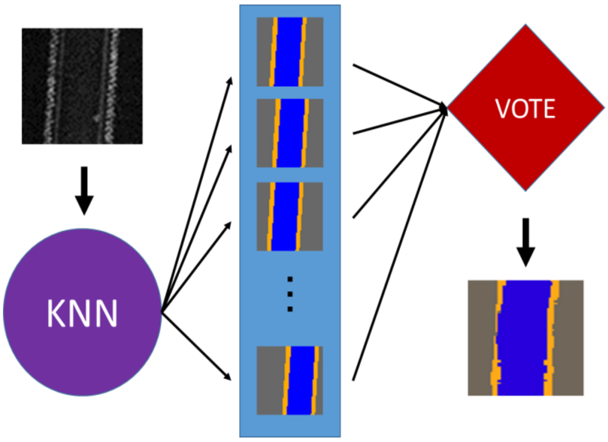

Figure 3. The overview of image labeling by KNN.

3. EXPERIMENTS

3.1. Dataset

In experiments, we use 36 LIDAR intensity images with ground truth labels obtained by the MMS. The size of the image is 2000 × 1500 pixels, and one pixel represents four centimetres. Therefore, each LIDAR intensity image covers 80 ×60 meters. Figure 4 shows the examples of our dataset. Those images include 9 categories (pedestrian crossing, catchment basins, roadside tree, gutter, gore area, road, median, pedestrian path and road shoulder). We divide 36 images into training (28 images) and test sets (8 images). We cropped semantic regions of various sizes from training and test images. Caffenet resize an image to 256×256 pixels automatically. Thus, we try to use 48×48, 64 ×64 and 96×96 pixels on the basis of 64×64 pixels that is quarter of 256×256 pixels.

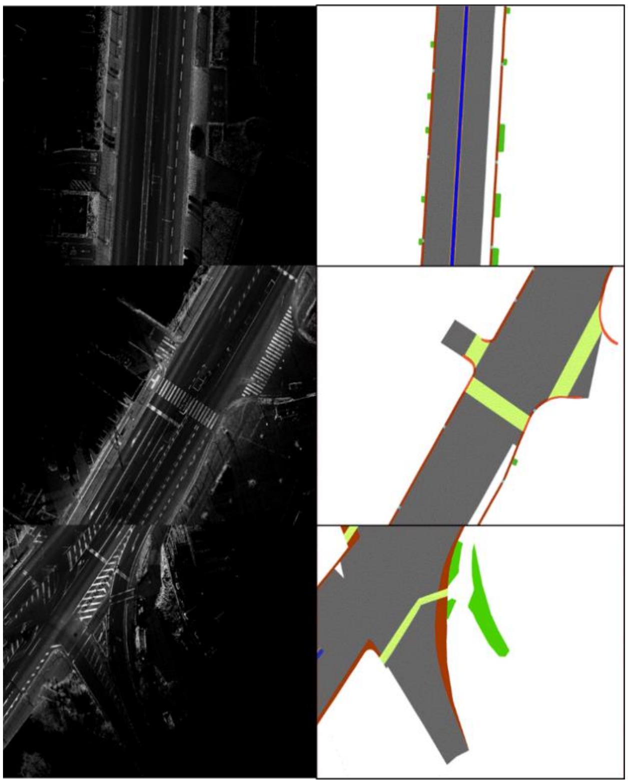

Figure 4. Example of our dataset. Left: LIDAR intensity image, Right: Ground truth label

3.2. Results

We use both class average accuracy and pixel-wise accuracy as evaluation measures. Class average accuracy is the average of classification accuracy of each class, and it is influenced by the classes with small area such as catchment basins. Pixel-wise accuracy is the percent of correctly labeled pixels in all pixels of test images, and it is influenced by the classes with large area such as road. We consider that class average accuracy is more important than pixel-wise accuracy because the purpose of this study is for making the Fundamental Geospatial Data of road automatically. Thus, it is better to recognize classes with small area such as catchment basins with higher accuracy. Parameter K in KNN is set to 2 by the experiment shown in Figure 5 because class average accuracy gave the best. A small K gave better accuracy than a large value because almost regions contain road.

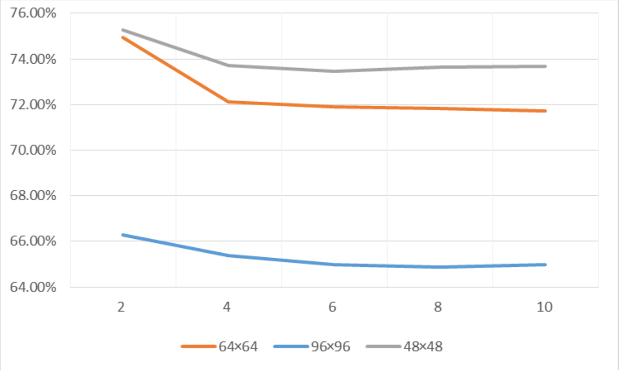

Figure 5. Class average accuracy while changing K. The horizontal axis represents the parameter K and the vertical axis represents the class average accuracy.

	48×48	64×64	96×96
Pixel-wise	96.30%	97.63%	97.13%
Class average	75.27%	74.96%	66.29%
pedestrian crossing	77.42%	75.67%	69.73%
catchment basins	28.23%	34.81%	7.26%
garden plant	93.70%	93.30%	87.60%
gutter	25.22%	18.48%	0.00%
gore area	74.93%	74.66%	62.47%
road	96.80%	98.75%	98.84%
median	96.31%	96.52%	94.49%
pedestrian path	97.32%	98.52%	97.84%
road shoulder	87.50%	83.95%	78.41%

Table 1. Accuracy of our method. Left: 48 ×48 pixels, middle: 64 ×64 pixels, Right: 96 ×96 pixels

Table 1 shows the accuracy at K=2. Our proposed method obtains 97.63% in class average accuracy and 74.96% in pixel-wise accuracy when we use regions with 64 ×64 pixels. Road, pedestrian path, roadside tree and road shoulder are classified with high accuracy. Surprisingly, we do not use the knowledge about structure of road. We just use the similarity of features obtained by CNN. The accuracy of catchment basins and gutter is quite low. It is difficulty to classify these classes because the region size of catchment basins is small. The size is just 16 x 12 pixels. Figure 7 shows the example of catchment basins in the test image. On the other hand, reason of low accuracy in gutter class is ambiguous boundary between road and gutter. Figure 8 shows the example of gutter.

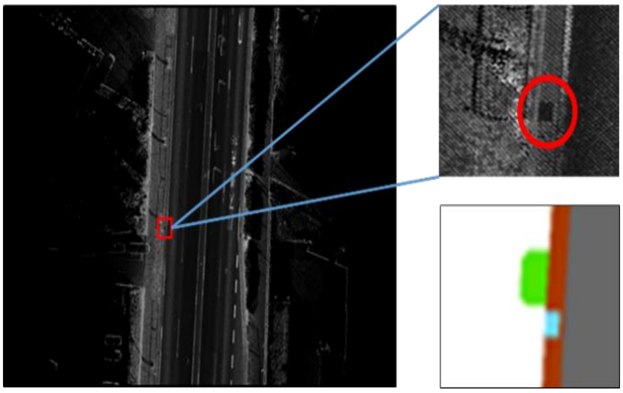

Figure 6. Example of catchment basins in the test image

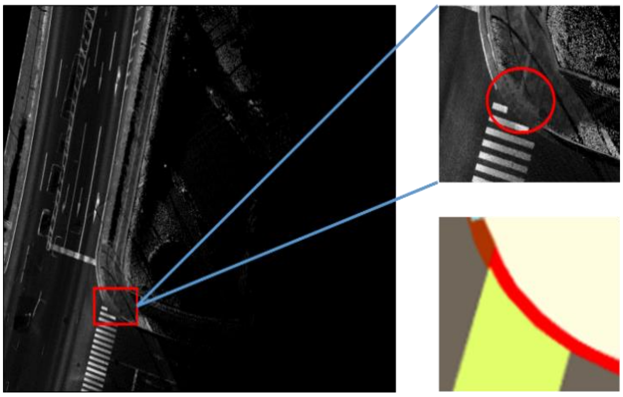

Figure 7. Example of gutter in the test image

Next, we combine the results with different region sizes. We find top K similar regions separately for each region size, and voting is performed independently. The class label with the maximum vote is assigned to each pixel. Table 2 shows the results when we combine the results with different region sizes.

The combination of 64×64 pixels and 48×48 pixels gave the highest class average accuracy. Table 2 shows that we can obtain high accuracy if we select appropriate region size. Figure 8, 9, and 10 shows our labeling result. Effectiveness of our method is demonstrated.

	only 48	48+64	48+64+96
Pixel-wise	96.30%	97.78%	98.06%
Class average	75.27%	75.53%	72.19%
pedestrian crossing	77.42%	76.43%	75.29%
catchment basins	28.23%	29.54%	14.55%
garden plant	93.70%	93.62%	91.57%
gutter	25.22%	19.46%	7.84%
gore area	74.93%	76.90%	75.52%
road	96.80%	98.56%	98.99%
median	96.31%	97.14%	97.02%
pedestrian path	97.32%	98.65%	99.01%
road shoulder	87.50%	89.53%	89.93%

Table 2. Accuracy of our method using different region sizes. Left: only 48 × 48 pixels. Middle: combination of 64 × 64 pixels and 48 × 48 pixels. Right: combination of all sizes.

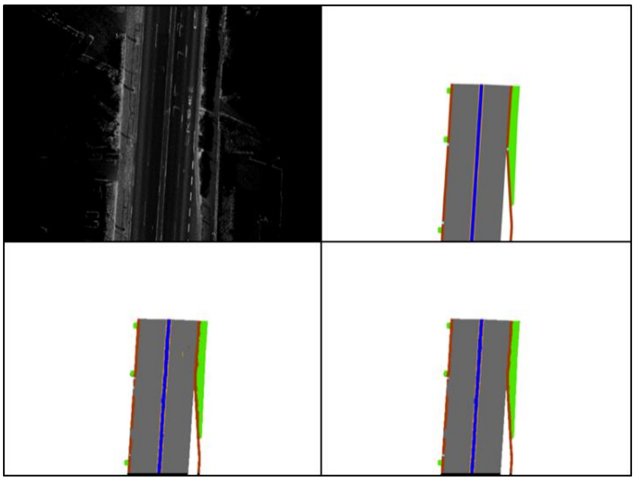

Figure 8. Example of our labeling results. Upper left: input image. Upper right: ground truth label. Lower left: only 48×48 pixels. Lower right: combination of 64×64 pixels and 48×48 pixels.

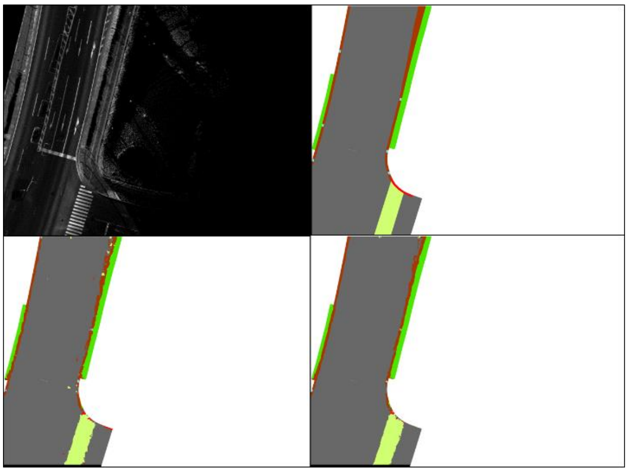

Figure 9. Example of our labeling results. Upper left: input image. Upper right: ground truth label. Lower left: only 48×48 pixels. Lower right: combination of 64×64 pixels and 48×48 pixels.

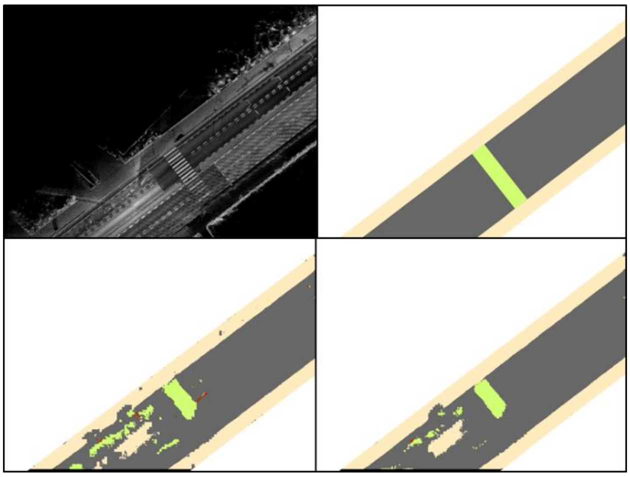

Figure 10. Example of our labeling results. Upper left: input image. Upper right: ground truth label. Lower left: only 48×48 pixels. Lower right: combination of 64×64 pixels and 48×48 pixels.

Figure 8, 9 shows that we can assign correct labels on almost all pixel. Figure 10 shows that labels of pedestrian crossing is cannot assign to ground truth position. The cause of this issue is blurred section of white line. There is no blurred white line in train images. Thus, we can resolve this issue if we can increase train images.

4. CONCLUSIONS

In this paper, we proposed the image labeling method for LIDAR intensity images obtained by MMS using the similarity of the feature obtained by CNN. Our method assigned the same label as human beings in 97.8% of the pixels in all test images. The results demonstrate that the proposed image labeling method is effective for LIDAR intensity images.

However, we must improve the accuracy of the classes with low accuracy such as catchment basis and gutter. We consider that each class has the adequate region size for extracting feature by CNN. For example, the feature catchment basis should be extracted from small region and the adequate size of cross walk should be large. Thus, we consider selecting the adequate region size for each class. This is a subject for future works.

ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI Grant Number 15k00252.

REFERENCES

Hasegawa, H. and Ishiyama, N., 2013. "Publication of The Digital Maps (Basic Geospatial Information)", Geospatial Information Authority of Japan, Vol.60, pp.19-24.

Novak, K., 1993. "Mobile Mapping System: new tools for the fast collection of GIS information," Proc. SPIE, Vol.1943.

LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. "Gradient-based learning applied to document recognition," Proceedings of the IEEE, Vol.86, No.11, pp, 2278-2324.

Dudani, S.A., 1976. "The distance-weighted k-nearest-neighbor rule," IEEE Transactions on Systems, Man and Cybernetics, Vol.4, pp.325-327.

Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, California, USA, pp. 1097-1105.

Jonathan, L., Shelhamer, E., and Darrell, T., 2015. "Fully convolutional networks for semantic segmentation." Proc. IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA, pp. 3431-3440.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S. and Darrell, T., 2014. "Caffe: Convolutional architecture for fast feature embedding", Proc. ACM International Conference on Multimedia, Florida, USA, pp. 675-678.

Deng, J., Dong, W., Socher, R., Li, L.J, Li, K. and Fei-Fei, L., 2015. "Imagenet: A large-scale hierarchical image database." Proc. IEEE Conference on Computer Vision and Pattern Recognition, Boston, USA, pp. 248-255.

Reyes, A.K., Caicedo, J.C., and Camargo, J.E., 2015. "Fine-tuning deep convolutional networks for plant recognition," Working notes of CLEF 2015, Toulouse, France, Vol.1943.