Aerial Image Segmentation via Noise Dispelling and Content Distilling

Yongqing Sun[1], Xiaomeng Wu[2], Yukihiro Bandoh[1], and Masaki Kitahara[1]

[1] NTT Computer and Data Science laboratories
[2] NTT Communication Science Laboratories
[3] yongqing.sun.fb@hco.ntt.co.jp
[4] yukihiro.bandoh.pe@hco.ntt.co.jp
[5] masaki.kitahara.ve@hco.ntt.co.jp
[6] xiaomeng.wu.px@hco.ntt.co.jp

Abstract. Aerial image segmentation is an essential problem for land management which can be used for change detection and policy planning. However, traditional semantic segmentation methods focus on single-perspective images in road scenes, while aerial images are top-down views and objects are of a small size. Existing aerial segmentation methods tend to modify the network architectures proposed for traditional semantic segmentation problems, yet to the best of our knowledge, none of them focus on the noisy information present in the aerial images. In this work, we conduct an investigation on the effectiveness of each channels of the aerial image on the segmentation performance. Then, we propose a disentangle learning method to investigate the differences and similarities between channels and images, so that potential noisy information can be removed for higher segmentation accuracy.

Keywords: Aerial image segmentation · Disentangle learning · Semantic segmentation.

1 Introduction

In the event of a large-scale disaster such as an earthquake or tsunami, it is necessary to quickly obtain information on a wide area in order to secure safe evacuation and rescue routes and to consider reconstruction measures. Aerial image segmentation is one of potential solutions. The result from aerial image segmentation can also be used to adjust governmental policy for subsidy, and to utilize the given resources for land management. In the past, this task was done manually and laboriously so it could only be done for a small number of photographs, which is not enough to capture the changes across large areas [5]. Moreover, such changes can be dramatic over time. Therefore, it is highly desirable to develop an automatic solution for the task.

A similar task known as semantic segmentation inputs an image and outputs a semantic segmentation map which indicates the class of an arbitrary pixel in the input image. For this task, various methods have been proposed using deep learning. Among them, CNN [13, 6, 12] has been actively applied in research and has shown high performance. However, they heavily rely on manual annotation because they are fully supervised [3, 1], yet remote-sensing data is not rich in training data, and it is not clear whether "deeper" learning is possible with a limited number of training samples.

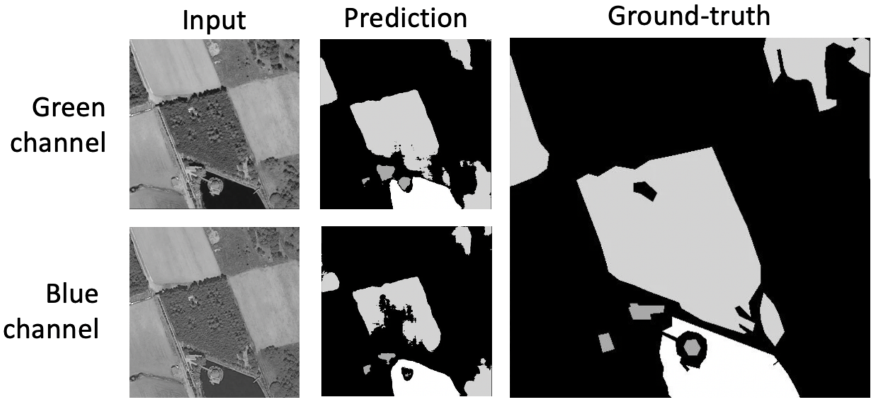

Fig. 1. U-Net results when trained on different channels show different effectiveness on different classes. The Green channel outperforms in the Woodland class (light gray) while the Blue channel outperforms in the Water class (white).

Semantic segmentation of low-resolution images in aerial photography is a challenging task because their objects are tiny and observed from a top-down faraway viewpoint, which is entirely different compared to the traditional semantic segmentation task. Many aerial segmentation methods are based on network architectures proposed for the mainstream semantic segmentation task. Boguszewski et al. [2] employs DeepLabv3+ with the backbone being modified Xception71 and Dense Prediction Cell (DPC) for aerial segmentation. Khalel et al. [8] proposed 2-level U-Nets with data augmentation to refine the segmentation result on aerial images. Li et al. [9] proposed to add a group of cascaded convolution to U-Net to enhance the receptive field.

However, aerial images are generally captured from high altitude and available in low resolution, hence their illumination information is sensitive to noise. The situation is worsen with the presence of dense objects. Therefore, in order to achieve accurate segmentation and classification of aerial photographs, it is necessary to extract the essential semantic content in the images.

Disentangle learning indicates methods which allows encoding the input into separated features belonging to predefined sub-spaces. UNIT [11] and MUNIT [7] are popular unsupervised disentangle learning methods for image-to-image translation task, which can be deemed as the general task of many computer vision problems such as segmentation. They encode the image into a content code and a style code. Content code is the common feature that can be shared between 2 domains, e.g. the same pose shared by a cat image and a tiger image. Style code is the encoded distribution which gives the variant tastes to the content code and it cannot be shared between 2 domains, e.g. the difference in skin color and texture between cat and tiger images. However, UNIT only allows one-to-one translation while MUNIT allows one-to-many translation with random style codes. Inspired by this technique, we propose the content consistency loss to enforce the content code of different channels of the same image to be the same and style consistency loss to enforce the style code of the same channel of different images to be the same.

In this work, we conducted segmentation on each channel respectively and found that each channel has different segmentation effectiveness on objects as shown in Figure 1. It is intuitively to think that different channels of the same image share the same content code, which is essential for aerial image segmentation. We also come up with a hypothesis that the same channels of different images share the same style code. The style code, which may reflect noise or specific illumination characteristic in channels, should not contribute to the segmentation result and has adverse effect on segmentation. Therefore, the content code should contain all the essential information needed for the segmentation task.

In our framework, there are 2 encoders and 3 generators. Each encoder and their corresponding generator is corresponding to an image channel. Taking the green channel as an example, the green channel encodes the green channel image into content code and style code, and the generator will generate back to the green channel image based on these two codes. We then define a reconstruction loss between two green channel images. The content consistency loss is added to ensure the content codes from green channel encoder and blue channel encoder to be the same for green channel images and blue channel images of the same image. The style consistency loss is also employed to ensure the style codes of the same channel is the same across different images. Finally, we use the content code of the two channels as the input to the segmentation generator followed with a semantic loss. The proposed method is evaluated on LandCover.ai dataset [2].

To summarize, our contribution is two-fold:

- We show that different channels have different focuses on different segmentation classes.
- We propose a disentangle learning framework that automatically remove the noisy information present in aerial images by mining the differences between channels of the same image and similarities of the same channel across different images.

The paper is organized as follows: Section 3 presents our proposed framework, section 4 shows the experimental results and detailed discussion, then our work is concluded with section 5.

2 Related work

2.1 Traditional semantic segmentation

Semantic segmentation task inputs an image and outputs a semantic segmentation map which indicates the class of an arbitrary pixel in the input image. However, it does not tell if those pixels belong to different instances. Traditional semantic segmentation focus on a road scene dataset such as Cityscapes [4], where big objects are generally available. DeepLabv3+ [6] proposes to use atrous separable convolution for the encoder. U-net [15] proposes skips which connect low-level features and high-level features.

On the other hand, objects in aerial images are captured from top-down views and in a very tiny form. Therefore, aerial segmentation presents challenges demanding approaches different from traditional semantic segmentation.

2.2 Aerial segmentation

Many aerial segmentation methods are based on network architecture proposed for the mainstream semantic segmentation task. A. Boguszewski et al. [2] employs DeepLabv3+ with the backbone being modified Xception71 and Dense Prediction Cell (DPC) for aerial segmentation. Khalel et al. [8] proposed 2-levels U-Nets with data augmentation to refine the segmentation result on aerial images. Li et al. [9] proposed to add a group of cascaded convolution to U-net to enhance the receptive field. However, none of them really focus on investigating the input aerial images for noise removal purpose.

We opt for U-net, which has a similar network architecture to FPN [10] for small object detection. The skip connections enhance semantic level of lower-level features and reduce information distortion at the input of high-level features. We do not employ FPN [10] directly as the size of the objects in the aerial images is already very small and that size does not vary much.

2.3 Disentangle learning

Disentangle learning indicates methods which allows encoding the input into separate features belonging to predefined sub-spaces. UNIT [11] and MUNIT [7] are popular unsupervised disentangle learning methods for image-to-image translation task, which can be deemed as the general task of many computer vision problems such as segmentation. They encode the image into two content code and style code. Content code is the common feature can be shared between 2 domains, e.g. cats and tigers may share the same pose. Style code is the encoded distribution which gives the variant tastes to the content code and it cannot be shared between 2 domains, e.g. skin colors and patterns between cats and tigers are not the same. However, UNIT only allows one-to-one translation while MUNIT allows one-to-many translation with random style codes.

Yet, due to their unsupervised nature, none of them can utilize the advantage of doing disentangle learning on separate channels of the same image: Given an

RGB image, we know for sure that their content code must be the same. We also take our constraint hypothesis further, the style code of an interested channel, e.g. green channel, should be the same across different images, which is currently not endorsed by any of the existing methods.

FCN+MLP [14] up-samples the encoded features of each layers in Base FCN and then combine them to yield the final prediction. Khalel et al. [8] proposed 2-levels U-Nets with data augmentation.

3 Proposed method

3.1 Framework

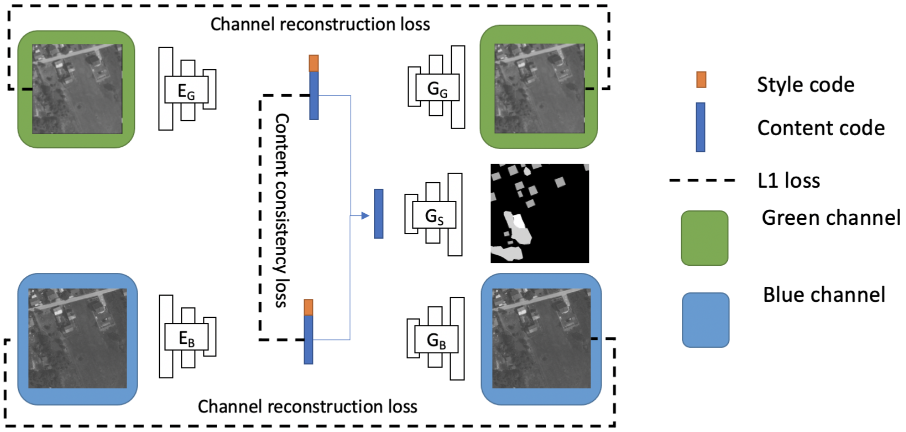

Fig. 2. (a) Style consistency loss

Let x_G, x_B be the G, B channels of an input image x. We exclude the R-channel from our framework as it is discovered to contain a lot of noisy information in Section 4.2.

We opt for U-net, which has a similar network architecture to FPN [10] for small object detection, as the network architecture of our encoders and generators. The skip connections enhance semantic level of lower-level features and reduce information distortion at the input of high-level features. We do not employ FPN [10] directly as the size of the objects in the aerial images is already very small and that size does not vary much.

As shown in Figure 3, our network consists of two encoders and three generators. The two encoders are G-channel and B-channel encoder; the three generators are G-channel generator, B-channel generator, and segmentation map generator. Let E_G be the G-channel encoder, $E_G(x_G)$ yields a 256-channel feature map, whose 56 first channels act as the style code s_{x_G} and 200 remaining

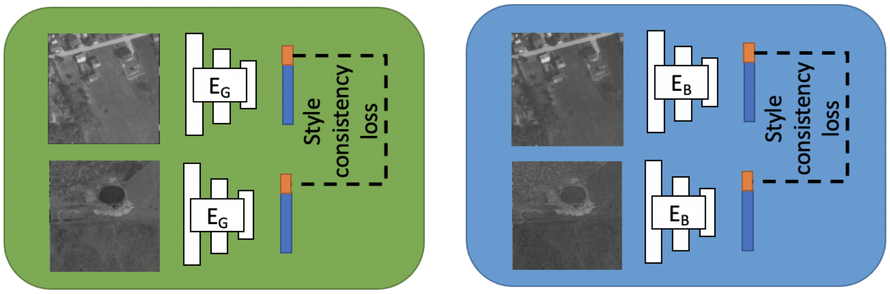

Fig. 3. (b) Content consistency loss and channel.
An overview of our proposed framework described with fig.(a) and (b). We employ U-net network architecture for our encoders E and generators G. Our network also uses a semantic segmentation loss, which is not shown for the sake of brevity.

channels act as the content code c_{x_G} as shown in Figure 3. The generator $G_G(c_{x_G}, s_{x_G})$ reconstruct the G-channel input while the generator $G_S(c_{x_G})$ predicts the semantic segmentation map.

Similarly, $E_B(x_B)$ yields the style code s_{x_G} and the content code c_{x_B}. We enforce the content code of G-channel c_{x_G} and B-channel c_{x_B} of the same image to be the same by the content consistency loss. This constraint is enforced further by training $G_S(c_{x_G})$ and $G_S(c_{x_B})$ to predict the same semantic segmentation map using the same network parameters G_S.

We argue that the style code of the same channel should be the same across different images, hence, while the style code is important for the channel reconstruction task, it should not contain any information useful for the segmentation task. Therefore, along with the style consistency loss between different images, the reconstruction generators G_G and G_B input the style code but the semantic segmentation map generator G_S does not.

3.2 Channel reconstruction loss

Let E_i and G_i be the i-channel encoder and i-channel reconstruction generator. Channel reconstruction loss enforces G_i to reconstruct the exact same channel which is the input of E_i.

$$L_r(x) = \sum_{i \in \{G,B\}} \|x_i - G_i(E_i(x_i))\|_1 \tag{1}$$

3.3 Content consistency loss

Let $(c_{x_i}, s_{x_i}) = E_i(x_i)$ be the content code c_{x_i} and style code s_{x_i} of the input i-channel. Content consistency loss enforces content codes of G-channel and B-channel of the same image to be the same.

$$L_c(x) = \|c_{x_G} - c_{x_B}\|_1 \tag{2}$$

One may concern that the content codes will converge to zero using this simple loss. However, the semantic segmentation loss will enforce them to be non-zero in order to generate a meaningful semantic segmentation map.

3.4 Style consistency loss

Style consistency loss, on the other hand, enforces the style codes of the same channel to be the same across different images. Let x and y be 2 different images, x_i and y_i be the i-channel of the two, style consistency loss can be computed as follows:

$$L_s(x, y) = \sum_{i \in \{G,B\}} \|s_{x_i} - s_{y_i}\|_1 \tag{3}$$

The style codes are also constrained to be non-zero by the channel reconstruction loss and content consistency loss. As the content codes are the same for all channels of the same image and the reconstructed channels should look differently, the style codes should be the main contributor of such differences, hence they are enforced to be non-zero.

If the batch size is larger than 2 then the style consistency loss will be computed on the first and second samples, second and third samples, etc. The style consistency loss is also applied on the first and last samples forming a cycle of style consistency.

3.5 Semantic segmentation loss

We use the weighted cross-entropy loss for the semantic segmentation task. Let M, N and K be the width, height and number of channels of the semantic segmentation map $z_i = G_S(x_i)$. Here, K is also the number of classes in the dataset and w_k is the loss weight of the class k. The larger w_k is, the more attention is spent on reducing the loss values from class k.

$$L_{ss}(x) = \tag{4}$$

$$-\sum_{i \in \{G,B\}} \sum_{m=1}^{M} \sum_{n=1}^{N} \sum_{k=1}^{K} w_k \log\left(\frac{e^{z_i(m,n,k)}}{\sum_{j=1}^{K} e^{z_i(m,n,j)}}\right)$$

(5)

3.6 Total batch loss

We set the task weight of the content consistency loss, style consistency loss as they are our fundamental constraint. We also set the task weight of the segmentation loss to 10 as it is our main task. Due to the nature of the style consistency loss, our total loss is provided in a form for the whole batch. Let $|B|$ be the batch size of a batch B and x_b be the b-th sample in the batch.

$$\begin{aligned} L(B) = & \tfrac{10}{|B|}\left(L_s(x_1, x_{|B|}) + \textstyle\sum_{b=2}^{|B|} L_s(x_{b-1}, x_b)\right) \\ & + \textstyle\sum_{b=1}^{|B|} (10 \times L_c(x_b) + L_r(x_b) + 10 \times L_{ss}(x_b)) \end{aligned} \tag{6}$$

(7)

4 Experiment result

4.1 LandCover.ai

To the best of our knowledge, LandCover.ai [2] is one of the benchmark aerial segmentation datasets that includes buildings, woodlands and water classes for our experimental purpose. They also has the background class, which involves objects and regions those do not belong to other 3 classes. It covers 216 km^2 of rural areas in Poland with the resolution being 25/50 cm/px. The training set has 7470 aerial images of size 512 x 512 pixels in the training set. The validation set and the test set have 1620 aerial images of the same size in each set.

We follow the previous work and use mIoU, which is the mean IoU of the four classes, as our evaluation metric.

$$IoU = \frac{TP}{TP + FN + FP} \tag{8}$$

Here, TP stands for True Positive, FN stands for False Negative, FP stands for False Positive.

The training dataset also suffers from serious class imbalance. To address this issue, we employ the weighted cross-entropy loss for the semantic segmentation task. The class distribution and our proposed class weights are shown in Table 1.

4.2 Investigation on RGB channels

We investigate the effectiveness of each channels on the segmentation result by training U-net on each channel separately. The result is shown in Table 2. As expected, Green channel has the best performance in Woodlands as it is the dominant color in the Woodlands class, Blue channel has the best performance in Water as it is the dominant color in the Water class. However, Red channel's

Table 1. Class distribution in the training set of LandCover.ai and our used class weight.

	Buildings	Woodlands	Water	Background
Percentage	0.86%	33.21%	6.51%	59.42%
Class weight	0.8625	0.025	0.1125	0.0125

Table 2. mIoU of U-net trained on separate channels.

Channel	Buildings	Woodlands	Water	Background	Overall
Red	60.6%	78.26%	62.63%	64.75%	66.59%
Green	79.61%	**90.06**%	90.47%	**88.46**%	87.15%
Blue	**80.6**%	89.74%	**91.05**%	88.2%	**87.4**%
Green & Blue	79.09%	90.56%	91.33%	88.64%	87.41%

performance is very poor, especially on the Buildings class which it is supposed to prevail as the roof color might be red. We suspect this is due to the lighting condition in this dataset. Therefore, in this work and for this dataset, we will only consider G-channel and B-channel. Our framework can be easily extended to a 3-channel version.

4.3 Aerial segmentation using disentangle learning on G-channel and B-channel

We initialize our network parameters with the pre-trained U-nets on G-channel and B-channel in the previous section where they are applicable. We train the network using Adam optimizer with weight decay set to 10^{-7}. As the optimization space of the network on this problem is very tricky, we first set the learning rate to 10^{-4} and then decrease in a phased manner to avoid overshooting.

As the training progresses, the content consistency loss and style consistency loss easily decrease down approximately to zero while the segmentation performance keeps increasing. This fact indicates that our hypothesis of style codes of the same channel across different images being the same is evidently reasonable.

We compared the proposed method with baseline implementation with DeepLabv3 in[2], which is non-augmentation version because we want to make clear performance comparison between two methods.The result is shown in table 3, which demonstrate that our method can disentangle semantic content successfully across G and B channels and their corresponding style noises, and shows better performance on image segmentation.

5 Conclusion

We conduct an investigation on the effectiveness of R, G, B channels on the segmentation performance and find that each channel especially performs well for

Table 3. mIoU of the proposed method and baseline implementation with DeepLabV3+OS4 in[2] .OS denotes encoder output stride during training and evaluation.

Methods	Buildings	Woodlands	Water	Background	Overall
DeepLabV3	77.53%	91.05%	93.84%	93.02%	88.86%
Our Methods	**79.47**%	**91.56**%	**94.33**%	92.64%	**89.5**%

different classes due to their dominant color present in those classes. We propose a disentangle learning method to remove potential noisy information by setting 2 important constraints: channels of the same image should share the same content code and the same channel in different images should share the same style code. Our method demonstrates the effectiveness on aerial image segmentation. In the future work, We will continue to do investigation on disentanglement of more channels such as hyperspectral images.

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence **39**(12), 2481–2495 (2017)
2. Boguszewski, A., Batorski, D., Ziemba-Jankowska, N., Zambrzycka, A., Dziedzic, T.: Landcover. ai: Dataset for automatic mapping of buildings, woodlands and water from aerial imagery. arXiv preprint arXiv:2005.02264 (2020)
3. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). pp. 801–818 (2018)
4. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
5. Gerard, F., Petit, S., Smith, G., Thomson, A., Brown, N., Manchester, S., Wadsworth, R., Bugar, G., Halada, L., Bezak, P., et al.: Land cover change in europe between 1950 and 2000 determined employing aerial photography. Progress in Physical Geography **34**(2), 183–205 (2010)
6. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
7. Huang, X., Liu, M.Y., Belongie, S., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018)
8. Khalel, A., El-Saban, M.: Automatic pixelwise object labeling for aerial imagery using stacked u-nets. arXiv preprint arXiv:1803.04953 (2018)
9. Li, X., Jiang, Y., Peng, H., Yin, S.: An aerial image segmentation approach based on enhanced multi-scale convolutional neural network. In: 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS). pp. 47–52. IEEE (2019)
10. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)

11. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. Advances in neural information processing systems **30**, 700–708 (2017)
12. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8759–8768 (2018)
13. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
14. Maggiori, E., Tarabalka, Y., Charpiat, G., Alliez, P.: Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In: 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). pp. 3226–3229. IEEE (2017)
15. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)