# Class-Imbalance Aware CNN Extension for High Resolution Aerial Image based Vehicle Localization and Categorization

Feimo Li [1,2], Shuxiao Li [1], Chengfei Zhu [1], Xiaosong Lan [1,2], Hongxing Chang [1]
[1] Institute of Automation, Chinese Academy of Science, Beijing, China
[2] University of Chinese Academy of Science, Beijing, China
e-mail: lifeimo2012, shuxiao.li, chengfei.zhu, lanxiaosong2012, hongxing.chang@ia.ac.cn

***Abstract*—High resolution aerial image based vehicle localization and categorization methods are crucial for many real life applications. Convolutional neural network based classifiers have already achieved very high performances, but are still suffering from the problem of class imbalance. To address this issue, an efficient bi-parted style network extension scheme based on a class-imbalance aware loss function is proposed. This novel loss function is devised by adding an extra class-imbalance aware regularization term to the normal softmax loss, and will force the feature maps in the extended network structure to be more sensitive to samples from the minority classes. This network extension is compared with its strong equivalent counter-parts in experiment, and comparably significant improvements on the minority classes can be observed.**

***Keywords-high resolution aerial image, vehicle detection, vehicle categorization, convolutional neural network, class-imbalance***

## I. INTRODUCTION

Compared to images from ground-set traffic cameras, aerial images have a greater continuous visual coverage, which makes them extremely helpful for applications such as large range traffic surveillance or multi-target tracking. The vehicle localization and categorization methods being developed based on them can be of essential importance for traffic flow structure analysis and other similar tasks. Despite of its greatness in coverage range, vehicles in a typical aerial image with ground sampling distance (GSD) around 0.13m are still very small, whereas a small private car can be only 7 to 9 pixels in width. This largely limits the amount of imagery information available for classification, and imposes a stricter constraint on the permitted deviation during localization. Because of this, any vehicle categorization result corresponded to a predicted position with deviation more than 4 pixels can regarded as erroneous. Thus in order to eliminate the cascaded error from the preceding localization in procedure, these two prediction processes are combined and performed in a joint manner by a convolutional neural network (CNN) based multi-class classifier. But such arrangement still cannot escape from the ubiquitous problem of class-imbalance, which actually means, in this article, having far more negative samples than the positive ones, and the quantities of vehicle belonging to different categories are highly imbalanced.

In the literature, the term vehicle localization is also often referred as vehicle detection when only position estimation is considered. The existing solely position targeted studies are numerous, and mainly base on two types of models: the explicit model and the implicit model. The implicit models generally refer to methods that locate vehicles or their components by using local features such as SIFT [1], LBP [2] or image objects [3]. Although can be be fast and efficient when tailored appropriately, they often have a more severe multi-detection side-effect which require a more challenging re-bundling process. Methods based on explicit model make detection via using fixed-shaped sliding window with comparable scale to the vehicle. By extracting features from that window, e.g. Haar [4], HOG [5], vehicle locations can thus be generated by sliding over the region of interest with pre-defined stepping pattern.

Unlike vehicle localization, studies on aerial image based vehicle classification are barely numerable [6,7,8]. All three studies isolate the localization and categorization procedures. Specifically, in [6], vehicles and their components are localized by image objects, and their types are predicted based on sizes of the objects. Locations in [7,8] are gotten using sliding window and categorization is done based on features including HOG, LBF and HF etc.

The problem of class imbalance is a common problem in classification, and many insightful articles and reviews have been accumulated on this field by far. The existing methods can be roughly divided into three categories: data-level, algorithm-level and the hybrid ones [9]. The data-level methods focus on alternating the distribution of training samples, balancing the sampling ratio from different classes during training. Representative methods include the SMOTE [10] and many of its variants. The algorithm-level methods mainly focus on alleviating the bias on majority classes by alternating the cost function to assign greater penalties on samples from the minority classes in training. Another important branch in the algorithm-level solutions is one-class learning, which focuses on improving classification performances on a single subset of samples. The hybrid ones mix the previous two to improve the sampling scheme and learning algorithm, maximize the efficiencies by taking both advantages.

In this article, the class-imbalance problem is addressed via bi-parted extension to a typical convolutional neural network (CNN), where a novel class-imbalance aware loss function is proposed to maximize the improvement efficiency on the minority classes from the extra structure. Experiments show significant improvements in minority classes with equivalent or even smaller sized network extensions. Finally, there are some similar works need to be mentioned. Unlike the other bi-parted extensions [11-13], the extension employed in this article mainly focuses on achieving a cost-effective extension with less overhead while more efficiency. The principle of our loss function modification is very similar to that in paper [14], but the proposed one in that paper is for binary classification problem, and cannot be directly applied to multi-class ones.

## II. The Softmax Loss Function and the Class-Imbalance Aware Loss Term

### A. *The Bi-Parted Network Extension and the Softmax Loss function*

As has been suggested in paper [15], the magic behind the power of CNN based classifier is representation learning, by which each convolutional kernels is formed as a robust model of a specific texture pattern modeling the samples in different classes. And by employing the varied valued positional activations produced by these kernels in their corresponding feature maps, the trailing fully-connected layers in the network make categorical predictions on the given image samples. Therefore, if more relevant kernels are provided and trained for samples belonging to the minority classes, their classification performances will be improved.

Following this logic, a straight-forward solution for improving the classification performance on those minority classes is to simply increase the number of kernels in the convolutional layers, especially the last one at the top of network with its generated feature maps having direct links to the hidden neurons in the trailing fully-connected layers. As has been shown in Fig. 1, in which a typical CNN structure with 5 convolutional layers is extended with 3 new convolutional layers E-CONV3, E-CONV4 and E-CONV5. This CNN structure is used as the baseline in the experiment analysis part in experiment, and the extension is treated as a strong counterpart for the improved efficient extension scheme proposed in this paper.

But the categorical assignment of these newly added kernels is determined by the behavior of the loss function being employed for network optimization, which in this case, the softmax loss is usually employed. The softmax loss is derived from the softmax function which transforms the vector values $\mathbf{z}$ from the fully-connected layers onto a Bernoulli probabilistic distribution. The definition for softmax function is given in Equation (1).

$$P\left(y=i \mid x\right)=\text{softmax}\left(z_i\right)=\frac{\exp\left(z_i\right)}{\sum_j \exp\left(z_j\right)} \tag{1}$$

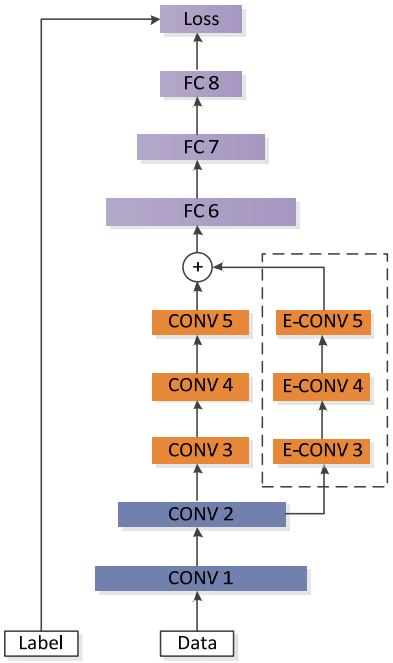

Figure 1. The straightforward network extension scheme.

In (1), $z_i$ is the categorical estimation value on class $i$ from outputted vector $\mathbf{z}$. During training, this log format of this probability function is taken as the loss function being imposed on the categorical label of the given sample, which is defined in Equation (2), and is minimized by gradient descend based method.

$$\begin{aligned} L\left(\hat{y}, y\right) &= -\log\left(\text{softmax}\left(\mathbf{z}\right)_i\right) \\ &= \log\sum\nolimits_j \exp\left(z_j\right) - z_i \end{aligned} \tag{2}$$

Where in Equation (2), $\hat{y}$ and $y$ are the predicted and true categorical label for the given image sample, and $\text{softmax}\left(\mathbf{z}\right)_i$ is the softmax value on the true class index, which is $i$ in this case. As can be seen by the definition of softmax loss function, it tends to give the same magnitude of penalties on all classes regardless of their majority or minority properties.

### B. *The Main-Side Loss Function for Class-Imbalance*

According to the analysis in previous section, the major problems in the simple extension scheme is the evenly updating propoerty of the original softmax loss. Based on this issue, a simple modification is performed in the hope of improving the minority class sensitivities of the kernels in the extended network structure. To achieve this goal, as being illustrated in Fig. 2, the extended network structure is completely isolated from the main body of the original network structure, where an independent one-layered fully-connected layer is assigned to it for generating the categorical probabilities from the newly added feature maps.

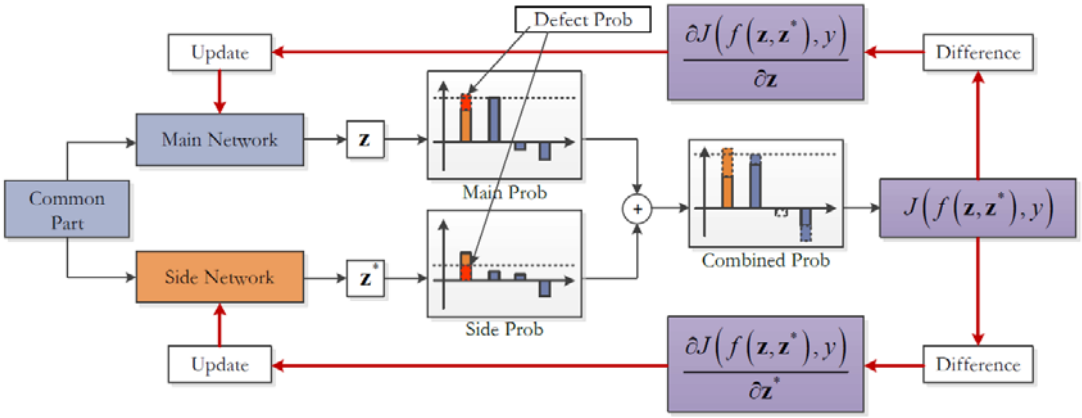


Figure 2. The principal structure of the Main-Side Network extension scheme.

For clarity, the original and extended network components are abbreviated as the Main and Side Networks. For the loss function being shown at the right-most of the figure, it is defined in Equation (3).

$$\begin{aligned} J\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right) &= L\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right)+\lambda\cdot\Omega\left(\mathbf{z}^*\right) \\ L\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right) &= -\log\left[\operatorname{softmax}\left(\mathbf{z}+\mathbf{z}^*\right)_{y=i}\right] \end{aligned} \tag{3}$$

In (3), $f\left(\mathbf{z},\mathbf{z}^*\right)$ is actually the summation $\mathbf{z}+\mathbf{z}^*$ of the categorical estimation values $\mathbf{z}$ and $\mathbf{z}^*$ from the Main and Side Network. $L\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right)$ is the softmax loss, and the term $\Omega\left(\mathbf{z}^*\right)$ is an extra penalty correlated solely with the output $\mathbf{z}^*$. This ensures the numerical differences between the updating differences for the Main and Side Networking used during training, with their back-propagation values are given in Equation (4).

$$\begin{aligned} \frac{\partial J\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right)}{\partial \mathbf{z}} &= \operatorname{softmax}\left(\mathbf{z}+\mathbf{z}^*\right)_{y=i}-\mathbf{1}\left(y=i\right) \\ \frac{\partial J\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right)}{\partial \mathbf{z}^*} &= \frac{\partial J\left(f\left(\mathbf{z},\mathbf{z}^*\right),y\right)}{\partial \mathbf{z}}+\lambda\cdot\frac{\partial\Omega\left(\mathbf{z}^*\right)}{\partial \mathbf{z}^*} \end{aligned} \tag{4}$$

As can be seen from (4), the updating difference for the Side Network is almost the same as that for the Main Network, except for the trailing $\lambda\cdot\frac{\partial\Omega\left(\mathbf{z}^*\right)}{\partial \mathbf{z}^*}$. In fact, the term $\Omega\left(\mathbf{z}^*\right)$ acts as a controlling factor to change the penalization values on different classes. Recalling that the softmax loss is to be minimized during training, $\Omega\left(\mathbf{z}^*\right)$ should be high on samples from majority classes and low on those from minority classes. This regulation can also be interpreted as to achieve the highest classification improvement with the minimal amount of probability rectification from the Side Network.

Base on this intuition, the controlling penalization $\Omega\left(\mathbf{z}^*\right)$ is defined as a weighted Norm-2 penalty for the categorical estimation outputs from the Side Network, as in Equation (5) and (6).

$$\Omega\left(\mathbf{z}^*\right)=\left\|\mathrm{B}\odot\mathbf{z}^*\right\|_2=\sqrt{\sum\nolimits_j\left(\beta_j\cdot z_j^*\right)^2} \tag{5}$$

$$\beta_j\propto ACC\left(\mathbf{X}\right)_j, ACC\left(\mathbf{X}\right)_j=\frac{TP\left(\mathbf{X}\right)_j}{TP\left(\mathbf{X}\right)_j+FP\left(\mathbf{X}\right)_j} \tag{6}$$

In (5), the operator $\odot$ represents the element-wise multiplication of vectors, and the $\|\cdot\|_2$ is the Norm-2 operation. The categorical penalization coefficient vector $\mathrm{B}=\left\{\beta_j\right\}$ is defined as the composition of the categorical classification accuracies measured by the Main Network, and the $\mathbf{X}$ is the set of input image samples on which class-wise accuracies are evaluated. Such definition ensures that the penalizations are greater on the majority classes which have already been well classified.

Moreover, since there are only output neurons equivalent to the number of classes in this fully-connected layer in the Side Network, less connections are needed to connect with those extended feature maps, by which parameters are saved.

### *C. The Three Penalization Schemes and ReLU Positive Constraint on the Side Network Probabilities*

Since the categorical penalization coefficients are based on the class-wise classification accuracies by the Main Network, there are several ways to calculate it by changing the obtaining scheme for the set of image samples $\mathbf{X}$. In this article, three major calculation schemes are analyzed, which are listed in Table. 1.

TABLE I. THE THREE PENALIZATION MODES

| Schemes | Descriptions |
|---|---|
| Global | Calculate the class-wise accuracies based on all the training samples by the original network. |
| Local | The training samples are firstly clustered in the probabilistic space by the original network, and local classification probabilities are calculated on the clusters, where the penalization coefficients will be adjusted locally according to this clustering scheme during training. |
| Batch-wise | Class-wise classification accuracies are calculated from the training sample mini-batch, which will change dynamically during the training. |

Both the global and local penalizations are calculated before the fine-tuning of the network.

These schemes represent typical accuracy evaluations performed either globally or locally, all has its own advantages and disadvantages. For the global version, it lacks enough flexibility for accuracy variances in the local probability space. The local version based on clustering partial overcomes this problem, but is still based on an accuracy evaluation executed before the fine-tuning process starts, and can become inappropriate as probability space changes during training. The batch-wise version keeps an alive tracking of the local accuracies but might be too flexible for a stable optimization.

Another important factor to consider is whether a ReLU layer should be used as the positive constraint on the likelihood values given by the Side-Network. Without such constraint, likelihoods from the Side Network can be regarded as a fluctuated adjustment with constrained magnitude. But when this ReLU layer exists, Side Network likelihood can be regarded as a positive increment on the estimations where deficiency in probability exists.

## III. EXPERIMENTAL RESULTS

### A. *The Data Set and the Networks Extension Counterparts*

The proposed extension scheme is compared with others on the Munich dataset [7]. It is an aerial image dataset with 20 high resolution aerial images at size of 5616 x 3744 with ground spatial definition (GSD) up to 13 cm. To facilitate the analysis and comparisons, uniformly sized image crops at 48 x 48 are manually extracted at all possible sliding window locations to as the training and testing samples. For these image patches, those having patch to vehicle center greater than 3 pixels are marked as negatives, while for the rest positive ones, four types of vehicles are considered in this study: sedan, station wagon, van and working truck. The quantity occupation ratios for them in the training set are 23.06%, 66.96%, 9.10% and 0.88%. The orientations for these positives are evenly quantized into 16 categories with 22.5 degree spacing, and all of them are rotated to the other 15 directions for data augmentation.

All experiments are done on a personal computer with i7 4970k CPU and a GTX 960 GPU based on the Caffe CNN computing framework. The baseline network (Orig.) used for extension and experimental analysis is VGG-M with 5 convolutional layers. The simple form of extension (Simple Ext.) is the same as in Fig. 1 as the three new untrained convolutional layers, where every one of them has 128 kernels. Whereas or the Main-Side Network bi-parted extension, an extra fully-connected layer is placed above E-CONV5 to produce the likelihood $\mathbf{z}^*$. Parameter and memory sizes of these extensions are listed in Table. 2, where the extension M-S Ext. has almost the same parameter size as the Orig., much smaller than Simple Ext. version. The memory consumption, on the other hand, is a little higher in the case of using Caffe implementation framework.

TABLE II. THE PARAMETER AND MEMORY COSTS FOR DIFFERENT NETWORK EXTENSION SCHEMES

| Size / Net | Orig. | Simple Ext. | M-S Ext. |
|---|---|---|---|
| Param. (Mb) | 361.7 | 439.6 | 362.2 |
| Mem. (Mb) | 1820 | 1988.4 | 1977.4 |

Parameter and memory costs are measured based on outputs from Caffe framework.

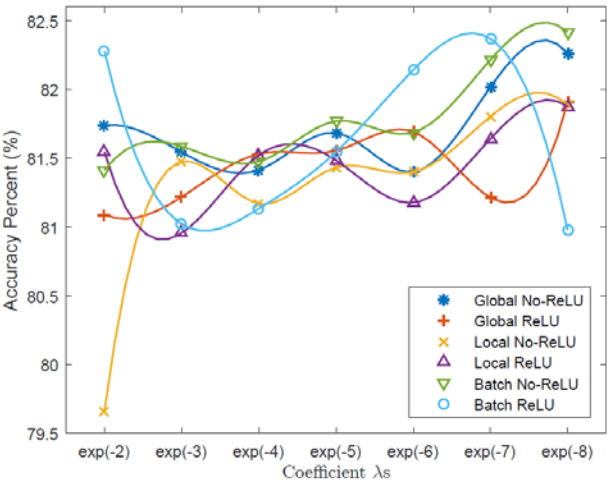


Figure 3. Influences of the coefficient on the averaged accuracies.

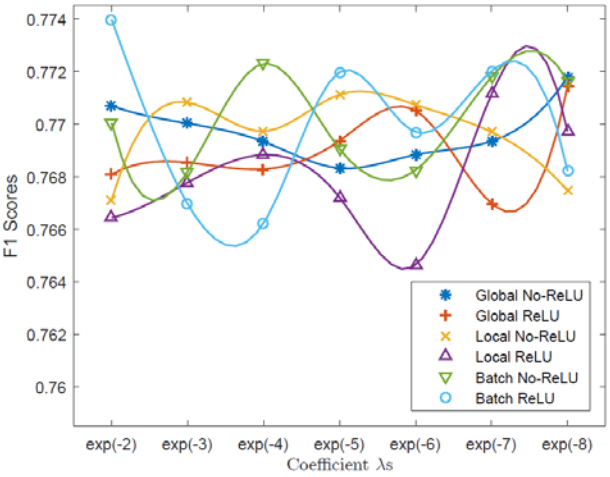


Figure 4. Influences of the coefficient on the averaged accuracies.

### B. *The Influences of the Coefficent $\lambda$ , The Tree Penalization Schemes and the ReLU Layer*

The influence of the penalization coefficient $\lambda$ on the averaged accuracies and the F1 scores is shown in Fig. 3 and Fig. 4. As can be observed from Fig.3, for all three penalization modes with or without the ReLU constraint, averaged accuracies increase as this coefficient $\lambda$ decreases. It is because the penalization on the Side Network likelihood goes weaker. In contrary, as in Fig. 4, the influence of $\lambda$ on F1 scores is less significant, as they all carry a fluctuation pattern around a fixed value. The influences on accuracies from the ReLU layer is also not significant, and it turns out only to strengthen the fluctuations on the F1 scores when it exists.

### C. *Extension Efficiency Compared to Strong Counterparts*

As being listed in Table. 3 and Table. 4, the proposed Main-Side Network Extension scheme (M-S Ext.) has achieved almost all the highest rank in accuracies and F1 scores, and the occurrences of the out-performances generally appear on the moderately sized minority classes, e.g. the Sedan, Van classes. Considering the M-S Ext. extension scheme has smaller parameter and memory consumption overheads, the proposed extension scheme is more cost-effective than the simple version Simple Ext..

TABLE III. COMPARISONS OF THE CLASS-WISE ACCURACIES BETWEEN COUNTERPART NETWORKS

| **Class / Net** | **Orig.** | **Simple Ext.** | **M-S Ext.** |
|---|---|---|---|
| Negative | 96.83% | 97.20% | **97.30%** |
| Sedan | 58.40% | 63.38% | **63.87%** |
| Station Wagon | **81.81%** | 82.13% | 81.67% |
| Van | 90.11% | 91.92% | **92.66%** |
| Working Truck | 69.72% | **74.52%** | 72.78% |

The highest and second highest values are marked by bold and underlines..

TABLE IV. COMPARISONS OF THE CLASS-WISE F1 SCORES BETWEEN COUNTERPART NETWORKS

| **Class / Net** | **Orig.** | **Simple Ext.** | **M-S Ext.** |
|---|---|---|---|
| Negative | 0.9791 | **0.9822** | 0.9820 |
| Sedan | 0.6247 | **0.6474** | **0.6474** |
| Station Wagon | 0.8010 | 0.8245 | **0.8273** |
| Van | 0.8422 | 0.8459 | **0.8505** |
| Working Truck | 0.5435 | **0.5666** | 0.5556 |

The highest and second highest values are marked by bold and underlines...

## IV. CONCLUSION

In this paper, a cost-efficient network extension scheme is proposed to address the issue of class-imbalance in the joint vehicle localization and categorization problem. The extended network component is trained by a novel class-imbalance aware loss function to be more sensitive to samples from the minority classes. The resulting extension has less parameter and memory consumption overhead and is capable of achieving equivalent or higher classification performances. Future work will be focused on feature map selection to further reduce the extra convolution cost from the newly added convolutional layers.

## ACKNOWLEDGMENT

This work is supported by National Science Foundation of China (NSFC) under grantings No. 61302154 and No. 61573350.

## REFERENCES

[1] Moranduzzo, T., Melgani, F.: Detecting cars in uav images with a catalog-based approach. IEEE Transactions on Geoscience and Remote Sensing 52 (2014) 6356–6367

[2] Moranduzzo, T., Mekhalfi, M.L., Melgani, F.: Lbp-based multiclass classification method for uav imagery. In: Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International, IEEE (2015) 2362–2365

[3] Tan, Q., Wang, J., Aldred, D.A.: Road vehicle detection and classification from very-high-resolution color digital orthoimagery based on object-oriented method. In: IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium. Volume 4. (2008) IV – 475–IV – 478

[4] Xu, Y., Yu, G., Wang, Y., Wu, X., Ma, Y.: A hybrid vehicle detection method based on viola-jones and hog + svm from uav images. Sensors 16 (2016)

[5] Tuermer, S., Kurz, F., Reinartz, P., Stilla, U.: Airborne vehicle detection in dense urban areas using hog features and disparity maps. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 6 (2013) 2327–2337

[6] Holt, A.C., Seto, E.Y., Rivard, T., Gong, P.: Object-based detection and classification of vehicles from high-resolution aerial photography. Photogrammetric Engineering & Remote Sensing 75 (2009) 871–880

[7] Liu, K., Mattyus, G.: Fast multiclass vehicle detection on aerial images. IEEE Geoscience and Remote Sensing Letters 12 (2015) 1938–1942

[8] Razakarivony, S., Jurie, F.: Vehicle detection in aerial imagery: A small target detection benchmark. Journal of Visual Communication and Image Representation 34 (2016) 187–203

[9] Krawczyk, B.: Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence 5 (2016) 221–232

[10] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research 16 (2002) 321–357

[11] Wang, L., Ouyang, W., Wang, X., Lu, H.: Visual tracking with fully convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 3119–3127

[12] Wang, N., Li, S., Gupta, A., Yeung, D.Y.: Transferring rich feature hierarchies for robust visual tracking. arXiv preprint arXiv:1501.04587 (2015)

[13] Marcu, A., Leordeanu, M.: Dual local-global contextual pathways for recognition in aerial imagery. arXiv preprint arXiv:1605.05462 (2016)

[14] Shen, W., Wang, X., Wang, Y., Bai, X., Zhang, Z.: Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3982–3991

[15] Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., Lipson, H.: Understanding neural networks through deep visualization. arXiv preprint arXiv:1506.06579 (2015)

## AUTHORS' BACKGROUND

| Your Name | Title* | Research Field | Personal website |
|---|---|---|---|
| Feimo Li | Phd candidate | Aerial Image Analysis | |
| Xiaosong Lan | Phd candidate | Aerial Video Analysis | |
| Shuxiao Li | full professor | Computer Vision | |
| Chengfei Zhu | full professor | Computer Vision | |
| Hongxing Chang | full professor | Artificial Intelligence | |

*This form helps us to understand your paper better, the form itself will not be published.

*Title can be chosen from: master student, Phd candidate, assistant professor, lecture, senior lecture, associate professor, full professor