Catfish density estimation by aerial images analysis and deep learning

Donatello Conte, Pierre Gaucher, Carlo Sansone

Catfish Density Estimation by Aerial Images Analysis and Deep Learning

Donatello Conte[a,*], Pierre Gaucher[a], Carlo Sansone[b]

[a] *Université de Tours, Laboratoire d'Informatique Fondamentale et Appliquée de Tours (LIFAT - EA 6300), 64 Avenue Jean Portalis, 37000 Tours, France*
[b] *Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione - Universitá degli Studi di Napoli Federico II - Via Claudio, 21 - Napoli - Italy*

ABSTRACT

The food economic chain of many rivers is based on an important control of the presence of predatory fish in the water. The assessment of the predation pressure on migratory species cannot be done manually. Therefore some automatic techniques are needed. In this paper we propose, for the first time, a deep neural architecture to estimate the catfish density from Aerial Images taken on the Loire river. The proposed architecture is adapted to the problem in hand and some variations to existing approaches, never applied in this application context, are designed to better fit the needs of the problem. Preliminary results show the appropriateness of the proposal and form the foundations for future developments on this new application.

1. Introduction

Recent studies in the Loire basin have shown that the catfish (*Silurus glanis*), known to be opportunistic from a food point of view, consume cyprinids but also migratory fish such as shad Boisneau (2015). This consumption in spring is not without impact of this species on the migratory fish community in the Loire and so impact to the economy linked to fish consumption.

Although it is not currently possible to assess predation pressure on migratory species, this pressure does exist. It would therefore be important to develop a method for estimating catfish densities in the natural environment, in the absence of obstacles, in order to begin to assess the predation pressure exerted by this species on migratory species including shad and salmon. It currently seems difficult to estimate catfish densities in the natural environment of the large river type without combining different techniques. Indeed, electric fishing can be used in habitats such as banks, plants, algae, for small individuals, and for depths of less than one metre but does not allow access to habitats such as large encumbrances or in areas with depths more than one meter. Underwater diving can be a complement for habitats in deep or difficult to access areas (blocks, crevices, encumbers, etc.) provided that the current is not a risk for the diver and that the transparency of the water allows observations. In very deep watercourses or canals, the use of multibeam sounders would have to be tested. Therefore, these two techniques are not applicable in all cases.

*Corresponding author
e-mail: donatello.conte@univ-tours.fr (Donatello Conte)

One promising technique is an estimation of catfish densities from aerial images analysis. This type of analysis is not very widespread. Several sections of the Loire have been filmed with a drone.

Several approaches were tested, in particular, indirect estimation techniques based on descriptor points (Harris' corners Derpanis (2004), SIFT Lowe (1999), SURF Bay et al. (2006)). These approaches have not been successful due to the very high variability of the images. and due to the presence of many artifacts (algae, trunks, vegetation, etc.). Because of these issues, more recent and probably more effective approaches are being considered in this paper for address this problem. Therefore, classification approaches based on deep neural networks (specifically convolutional neural networks, CNN LeCun et al. (2015)) have been applied. Deep neural networks, by their architecture, make it possible to avoid choosing the best descriptors to use for the estimation of the number of objects of interest and, on the contrary, they learn about the examples, the best configuration and the best parameters adapted to images to have an effective recognition.

Event CNN are now widely used in many computer vision taks, for the considered applicative context, there are several specificities that make this study interesting and original. Therefore the contributions of this paper are the following: first, it is the first time that image analysis techniques, and in particular CNN, are used in the context of catfish density estimation; more important, deep learning has been used mainly for classifications tasks, and less for regression tasks, image density estimation. Actually, even if there are some, few, approaches for people crowding estimation with CNN (as we will see in the next Section), this problem is still largely open and never addressed in the case of estimation density of different kind of objects than people. This make this paper the first attempt in this direction and lays the foundations for future developments.

The remainder of the paper is organized as follows: Section 2 discuss about related works, in particular the use of CNN for people crowding estimation; in Section 3 we describe the proposed architecture and some implementation details of our network while in Section 4 results of the application of the proposed techniques are drawn; Section 5 concludes the paper with some finally remarks and future perspectives.

2. Related Works

The first attempt to tackle the object counting problem has been called counting by detection (Barinova et al. (2012); Desai et al. (2011); Descombes et al. (2009); Dong et al. (2007); Moosmann et al. (2007)). The main idea is to use some detectors, to localize individual object instances in the image. Given the localization of all instances, counting becomes trivial. However, object detection is very far from being solved Borji et al. (2015), especially for overlapping instances. In our case, there are many overlapping instances, because catfishes usually move in group and many parts of single instances are not visible.

The second category of works is called counting by regression. These methods avoid solving the hard detection problem. Instead, a direct mapping from some global image characteristics (interest points, HOG, etc.) to the number of objects is learned

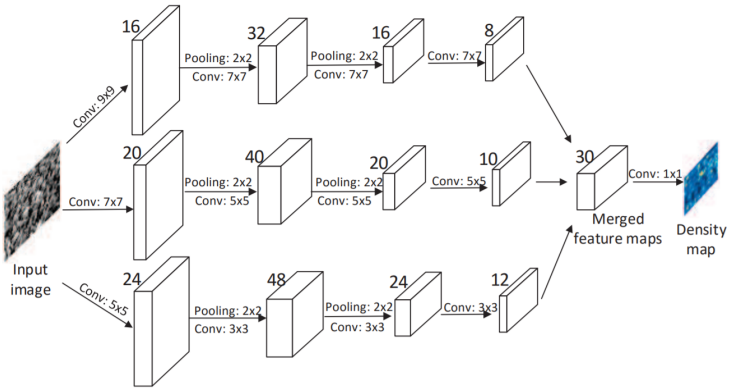

Fig. 1. The structure of network proposed in (Zhang et al., 2016).

(Cho et al. (1999); Kong et al. (2006); Conte et al. (2013); Zhang and Li (2012)). This approach however require a large number of training images with the supplied counts needs to be provided during training. This is not the case of our application framework as we will discussed later.

Recently, a more effective technique is based on learning density map model. They introduce a counting approach, which works by learning a linear mapping from local image features to object density maps. With a successful learning, one can provide the object count by simply integrating over regions in the estimated density map (Onoro-Rubio and López-Sastre (2016); Zhang et al. (2015, 2016); Boominathan et al. (2016)).

Authors in Zhang et al. (2015) propose a Convolutional Neural Network (CNN) based framework for cross-scene crowd counting. After a CNN is trained with a fixed dataset, a data-driven method is introduced to fine-tune (adapt) the learned CNN to an unseen target scene, where training samples similar to the target scene are retrieved from the training scenes for fine-tuning. This adaptation improve accuracy on a specific dataset, but it requires a fine-tuning phase each time an unseen target scene comes up.

The first work that introduce the dealing with the perspective problem in crowding estimation with Deep Neural Networks, is the work by (Zhang et al., 2016). The authors propose a multi-column convolutional neural network (MCNN) (inspired by the work of Cireşan et al. (2012)) containing three columns of convolutional neural networks whose filters have different sizes to take into account perspective problems. Then final predictions are obtained by averaging individual predictions of all deep neural networks. Figure 1 shows their proposed architecture.

Boominathan et al. Boominathan et al. (2016) propose the use of a combination of deep and shallow, fully convolutional networks to predict the density map for a given crowd image. Such a combination is used capturing both the high-level semantic information (face/body detectors) and the low-level features (blob detectors), that are necessary for crowd counting under large scale variations. Furthermore, they use a different augmentation data management on the different part of the network according to the available amount of training data at different scales. Figure 2 shows their proposed architecture.

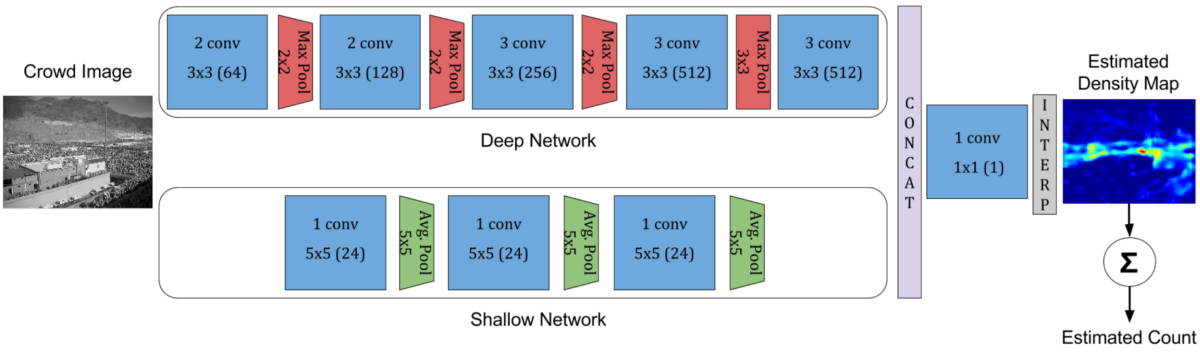

Fig. 2. The structure of network proposed in (Boominathan et al., 2016).

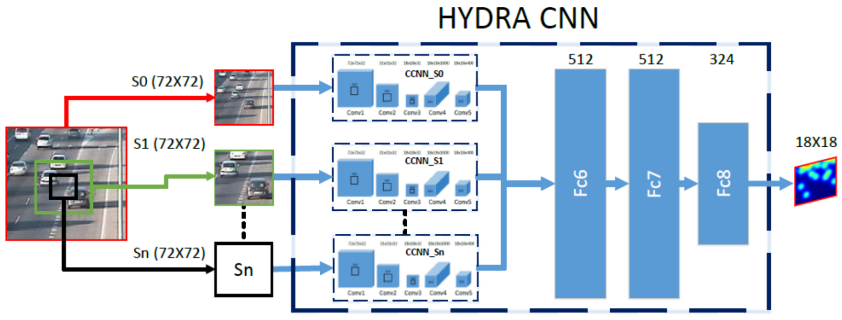

Fig. 3. The structure of network proposed in (Onoro-Rubio and López-Sastre, 2016).

Lastly, the authors of Onoro-Rubio and López-Sastre (2016) propose a new solution whose main contribution is that object densities can be estimated without the need of any perspective map or other geometric information of the scene. They do that by introducing the Hydra CNN architecture, a scale-aware model, which works learning a multiscale regressor inspired by the so called pyramidal network Lin et al. (2017). The problem of this approach remain the complexity of the network that is not necessary in our case in which we have limited scaling problems. Figure 3 shows their proposed architecture.

Our proposition is mainly inspired by this last work, but it present two main differences, which are also the contributions of this paper:

- First, since for aerial images there is not the problem of perspective, we simplify the deep network in such a way that it has not necessarily be aware of this issue. This results in two advantages: one is that the network is more efficient in terms of time processing; second there less parameters to learn so it needs less amount of training data.

- Second, as we will see our problem is very unbalanced, we change the data augmentation strategy in order to tackle this issue.

Furthermore, we want to highlight, that this proposal is the first one to address the problem of catfish density estimation by image analysis and deep learning.

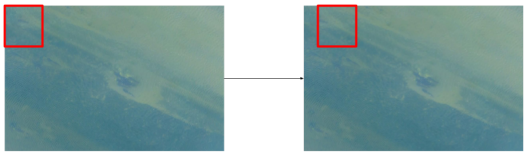

Fig. 4. The process of patches extraction.

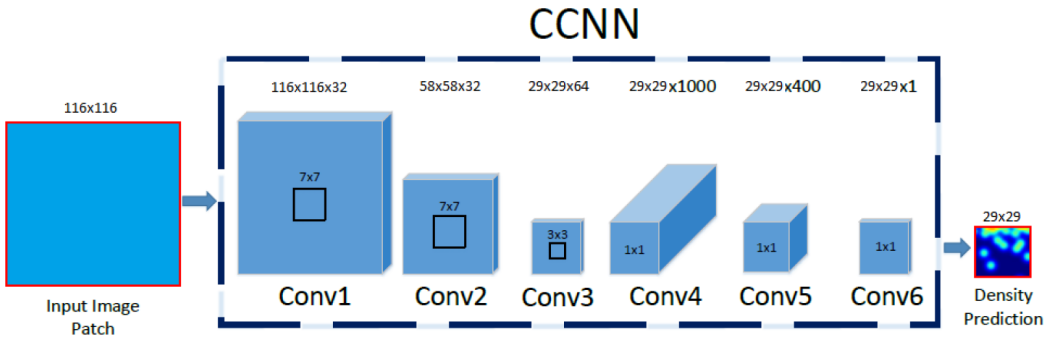

Fig. 5. Our proposal for the structure of deep network.

3. Proposed Framework

Figure 5 shows the proposed architecture. As you can see, it is inspired by HydraNet Onoro-Rubio and López-Sastre (2016), but it simplified to only one scale layer because Aeriel Images does not present the problem of perspective. The architecture is a classic ConvNet with 6 convolution layers interlaced by two max-pooling layers on the first two convolution levels. The input of the network is an image of 116x116 pixels size and the output is a density map (of 29x29 pixels size) of the input image.

For the problem in hand, we have hard unbalancing data. In fact we own a dataset of around 300 images of size 6000x4000, but only 12 images contain positive samples, i.e. there are catfishes within. Therefore, to address this unbalancing problem, we proceed as follows.

The first step is to extract patches for our big size images. In fact, as we said, the input size of the network is 116x116 and we have images whose size is 6000x4000. Therefore, we extract some 116x116 patches from the images starting from the top left corner and moving right a certain number of pixels (*stride*) and down when we reach the rightest limit of the image (see Figure 4). Now it is important to highlight that this extraction is not done in the same way on negative images (i.e. images without catfishes) and positive images. On positive images we used a dense extraction (stride size equals to 10). It is worth noting that on positive images, there are also parts of images without the object of interest, so with such a dense extraction we collect also negative samples. Therefore from negative images we extract, randomly, only 2 patches. This is done in order to deal with the unbalancing issue.

The second step is data augmentation, because we have very few positive samples. The augmentation is done by increasing the number of patches by making flips and rotations: in this way we obtain 21565 patches.

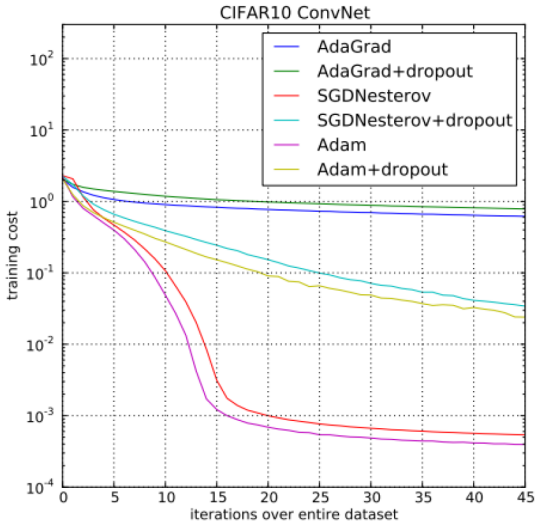

Fig. 6. Convolutional neural networks training cost for CIFAR10 deep architecture over 45 epochs (from Kingma and Ba (2015)).

It must be noted that the size of each patch (both original and density patches) is $116x116$ but the output results to be $29x29$ therefore we have to re-size the output density patches. This re-sizing has to be done carefully: when the sum of the pixels is made to obtain the estimation of the number of objects, this value has to be the same as the value provided by the network. Therefore, the re-sizing is done as follows: first, a uniform normalization in range $[0, 1]$ is performed; then the image is transformed to original input size; third, the values are re-normalized in order to give the expected value.

Concerning the implementation details of the network (following Onoro-Rubio and López-Sastre (2016) with some changes for adapting to our problem): the architecture consists of 6 convolutional layers; Conv1 and Conv2 layers have filters of size $7x7$ with a depth of 32, and they are followed by a max-pooling layer, with a $2x2$ kernel size; the Conv3 layer has $3x3$ filters with a depth of 64, and it is also followed by a max-pooling layer with another $2x2$ kernel; Conv4 and Conv5 layers are made of $1x1$ filters with a depth of 1000 and 400, respectively (fully convolutional architecture Long et al. (2015)); all the previous layers are followed by rectified linear units (ReLU); finally, Conv6 is another $1x1$ filter with a depth of 1, Conv6 is in charge of returning the density map estimation for the input patch P.

We adopt Adam algorithm Kingma and Ba (2015) for gradient-based optimization of stochastic objective functions. Empirical results demonstrate that Adam works well in practice and compares favorably to other stochastic optimization methods (see Figure 6).

3.1. Counting model and Ground Truth construction

It is important to spend some words to describe the process of the construction of the Ground Truth. In fact, while in classification and detection contexts, the ground truth for an image of video is straightforward, in case of density map estimation is somehow complex.

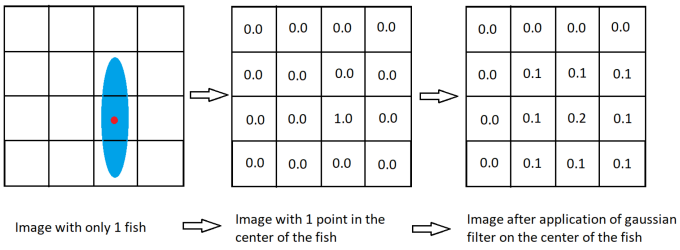

Fig. 7. A toy example to explain the process of ground truth annotation.

Our counting objects model follow the basic principles introduced by Lempitsky and Zisserman in Lempitsky and Zisserman (2010). Our solution requires a set of annotated images, where all the objects are marked by dots. In this scenario, the ground truth density map D_I, for an image I, is defined as a sum of Gaussian functions centered on each dot annotation,

$$D_I(p) = \sum_{\mu \in A_I} \mathcal{N}(p; \mu, \Sigma) \tag{1}$$

where A_I is the set of $2D$ points annotated for the image I, and $\mathcal{N}(p; \mu, \Sigma)$ represents the evaluation of a normalized $2D$ Gaussian function, with mean μ and isotropic covariance matrix Σ, evaluated at pixel position defined by p. With this density map DI, the total object count N_I can be directly obtained by integrating the density map values in D_I over the entire image, as in Eq. 2.

$$N_I = \sum_{p \in I} D_I(p) \tag{2}$$

Note that all the Gaussian are summed, so the total object count is preserved even when there is overlap between objects. Figure 7 shows a toy example of the annotation model and Figure 8 shows a real example of annotated image from our dataset.

4. Results

4.1. Our dataset

The Aerial Images which constitute our dataset show the considerable difficulty of the problem: the environment is highly variable in terms of colors, lightning, presence of obstacle and so on (see Figure 9); the appearance of the catfish is sometimes very similar to the background (see Figure 10) and it is difficult, even for a human expert to provide the real number of objects of interest also due to grouping of fishes; finally the presence of fishes is very sparse (often on an image of $6000x4000$ pixels there is only one catfish whose size is around $50x150$). In the dataset there 300 images, of which 12 positive images. We trained the network with 9 positive images and 3 negative images, then we test on the on the remaining 3 positive images.

In the test phase, for each image, we extract the patches without overlapping (stride equals to 0, we obtain the density map of the patch resulting from the application of our deep architecture on the patch, we obtain the number of the catfishes by sum all pixels values of the density map of the patch and the total number estimated of catfishes on the image is the sum of the values on all the patches of the image.

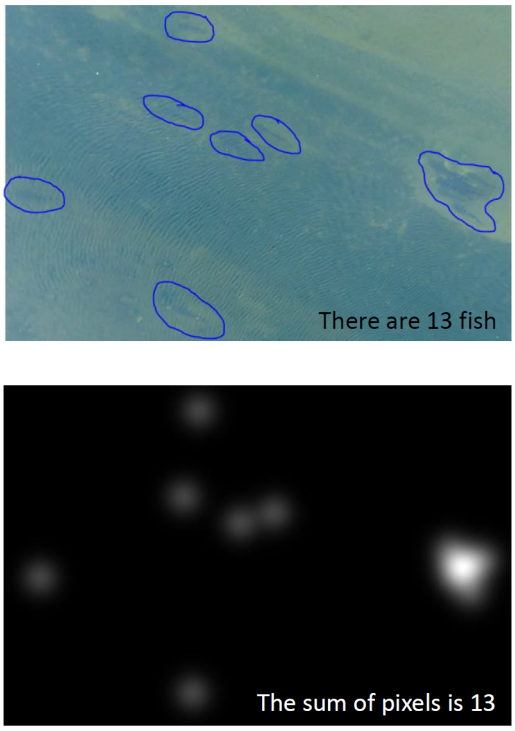

Fig. 8. A positive image sample and its corresponding ground truth annotation.

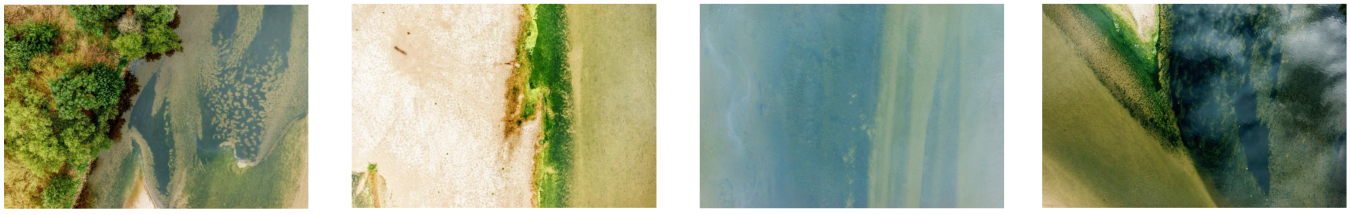

Fig. 9. Some example images from the considered dataset.

Fig. 10. An example of image in which the presence of catfishes is annotated by bounding boxes. Note the difficult for counting the group of catfishes on the right of the image.

Image	Estimation	Ground-truth
S1/DSC09754.jpg	101.97	2
S1/DSC09755.jpg	99.03	2
S1/DSC09768.jpg	104.23	13
S1/DSC09769.jpg	90.03	14
S1/DSC09770.jpg	96.45	1
S1/DSC09771.jpg	96.33	1
S1/DSC09773.jpg	88.60	1
S2/DSC00403.jpg	219.63	2
S3/DSC00592.jpg	102.89	1
MAE	106.92	
MSE	13017.92	

Table 1. Results on training set at epoch 0.

Image	Estimation	Ground-truth
S1/DSC09777.jpg	85.63	2
S2/DSC00404.jpg	128.74	2
S2/DSC00493.jpg	91.51	1
MAE	100.29	
MSE	10416.86	

Table 2. Results on test set at epoch 0.

4.2. Numerical Results

By following the convention of existing works Conte et al. (2010, 2013); Zhang et al. (2015) for crowd counting, we evaluate different methods with both the absolute error (MAE) and the mean squared error (MSE), which are defined as in the Eq. 3 and Eq. 4.

$$MAE = \frac{1}{N}\sum_{1}^{N}|z_i - \hat{z}_i| \tag{3}$$

$$MSE = \sqrt{\frac{1}{N}\sum_{1}^{N}(z_i - \hat{z}_i)^2} \tag{4}$$

where N is the number of test images, z_i is the actual number of people in the ith image, and $\hat{z}_i$ is the estimated number of objects (in our case catfishes) in the i^{th} image. Intuitively, MAE indicates the accuracy of the estimates, and MSE indicates the robustness of the estimates.

In this section we present the results obtained at different epochs of training, in order to understand the behavior of the network and to interpret the result in our application context. For each epoch, we show the number of fishes estimated on each training and test images and the MAE and MSE indexes on the entire training and test set.

Table 1 and Table 2 show results at epoch 0. Obviously the estimation is very bad, because the network has not yet learned.

After only 10 epochs the estimated values have suddenly decreased. For 7 out of 12 images, the estimation are below the true value (Table 3 and Table 4). The MAE and MSE continue to decrease at epoch 20.

Image	Estimation	Ground-truth
S1/DSC09754.jpg	1.08	2
S1/DSC09755.jpg	0.34	2
S1/DSC09768.jpg	4.83	13
S1/DSC09769.jpg	154.32	14
S1/DSC09770.jpg	3.00	1
S1/DSC09771.jpg	3.25	1
S1/DSC09773.jpg	5.04	1
S2/DSC00403.jpg	1.26	2
S3/DSC00592.jpg	0.58	1
MAE	17.83	
MSE	2198.66	

Table 3. Results on training set at epoch 10**.**

Image	Estimation	Ground-truth
S1/DSC09777.jpg	13.98	2
S2/DSC00404.jpg	28.43	2
S3/DSC00493.jpg	59.87	1
MAE	32.42	
MSE	1436.06	

Table 4. Results on test set at epoch 10**.**

Image	Estimation	Ground-truth
S1/DSC09754.jpg	13.89	2
S1/DSC09755.jpg	12.46	2
S1/DSC09768.jpg	13.50	13
S1/DSC09769.jpg	12.14	14
S1/DSC09770.jpg	11.88	1
S1/DSC09771.jpg	12.31	1
S1/DSC09773.jpg	12.26	1
S2/DSC00403.jpg	33.68	2
S3/DSC00592.jpg	18.02	1
MAE	11.87	
MSE	213.59	

Table 5. Results on training set at epoch 20**.**

Image	Estimation	Ground-truth
S1/DSC09777.jpg	5.07	2
S2/DSC00404.jpg	5.53	2
S2/DSC00493.jpg	8.19	1
MAE	4.60	
MSE	24.56	

Table 6. Results on test set at epoch 20**.**

Fig. 11. The estimated density map of Figure 10.

Image	Estimation	Ground-truth
S1/DSC09754.jpg	0.05	2
S1/DSC09755.jpg	0.02	2
S1/DSC09768.jpg	3.18	13
S1/DSC09769.jpg	0.00	14
S1/DSC09770.jpg	$4.38 \cdot 10^{-4}$	1
S1/DSC09771.jpg	$3.35 \cdot 10^{-5}$	1
S1/DSC09773.jpg	$7.82 \cdot 10^{-4}$	1
S2/DSC00403.jpg	0.01	2
S3/DSC00592.jpg	$7.2 \cdot 10^{-6}5$	1
MAE	4.10	
MSE	42.33	

Table 7. Results on training set at epoch 40**.**

Still at epoch 40, MAE and MSE continue to decrease, but actually all the estimation fall down to 0. The network weights seem to converge towards 0. This is mainly due to the sparsity of catfishes in the images. Even if we only took positive images for training, these are very big (6000x4000) with often only 1 positive sample (a catfish) in the image. Therefore, most of the 116x116-sized patches will not contain any fish. As a result, the neural network must return density maps composed entirely of zeros for a large part of the image (Figure 11 shows the estimated density map of the image depicted in Figure 10). This partly explains why the weights of the neural network seem to be converging towards zero.

Image	Estimation	Ground-truth
S1/DSC09777.jpg	$2.42 \cdot 10^{-5}$	2
S2/DSC00404.jpg	0.002	2
S2/DSC00493.jpg	0.032	1
MAE	1.65	
MSE	2.97	

Table 8. Results on test set at epoch 40**.**

5. Conclusions

In this paper we have presented a deep neural network for estimating the density of catfishes from aerial images. This is the first attempt to use machine learning and image analysis in this application context. The architecture has been inspired by the recent works on estimating people crowd density from images. But theses architecture was adapted at the specificity of the acquisition technology (aerial images) and the application context (catfish detection).

Preliminary results are not yes so good, the analysis shows many promising directions for improving performance. Future works will be precisely dedicated to improve the system in several ways: first we plan to study more in-depth the problem of data augmentation especially for positive sample, given their scarcity in our dataset; second we plan to use transfer learning technique to compensate for this lack of data; third we will try to add some application specific features that should improve the learning capability of the network.

References

Olga Barinova, Victor Lempitsky, and Pushmeet Kholi. 2012. On detection of multiple object instances using hough transforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 9 (2012), 1773–1784.

Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In *European conference on computer vision*. Springer, 404–417.

C. Boisneau. 2015. *Suivi des aloses en Loire moyenne et approche de la prédation par le silure (Monitoring of aloses in Loire and predation approach by the catfish)*. Technical Report. University of Tours.

Lokesh Boominathan, Srinivas SS Kruthiventi, and R Venkatesh Babu. 2016. Crowdnet: A deep convolutional network for dense crowd counting. In *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 640–644.

Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li. 2015. Salient object detection: A benchmark. *IEEE transactions on image processing* 24, 12 (2015), 5706–5722.

Siu-Yeung Cho, Tommy WS Chow, and Chi-Tat Leung. 1999. A neural-based crowd estimation by hybrid global learning algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 29, 4 (1999), 535–541.

Dan Cireşan, Ueli Meier, and Jürgen Schmidhuber. 2012. Multi-column deep neural networks for image classification. *arXiv preprint arXiv:1202.2745* (2012).

Donatello Conte, Pasquale Foggia, Gennaro Percannella, Francesco Tufano, and Mario Vento. 2010. A method for counting people in crowded scenes. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*. IEEE, 225–232.

Donatello Conte, Pasquale Foggia, Gennaro Percannella, and Mario Vento. 2013. Counting moving persons in crowded scenes. *Machine vision and applications* 24, 5 (2013), 1029–1042.

Konstantinos G Derpanis. 2004. The harris corner detector. *York University* (2004).

Chaitanya Desai, Deva Ramanan, and Charless C Fowlkes. 2011. Discriminative models for multi-class object layout. *International journal of computer vision* 95, 1 (2011), 1–12.

Xavier Descombes, Robert Minlos, and Elena Zhizhina. 2009. Object extraction using a stochastic birth-and-death dynamics in continuum. *Journal of Mathematical Imaging and Vision* 33, 3 (2009), 347–359.

Lan Dong, Vasu Parameswaran, Visvanathan Ramesh, and Imad Zoghlami. 2007. Fast crowd segmentation using shape indexing. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 1–8.

Diederik P Kingma and Jimmy Lei Ba. 2015. Adam: Amethod for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Dan Kong, Douglas Gray, and Hai Tao. 2006. A viewpoint invariant approach for crowd counting. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, Vol. 3. IEEE, 1187–1190.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436.

Victor Lempitsky and Andrew Zisserman. 2010. Learning to count objects in images. In *Advances in neural information processing systems*. 1324–1332.

Tsung-Yi Lin, Piotr Dollár, Ross B Girshick, Kaiming He, Bharath Hariharan, and Serge J Belongie. 2017. Feature Pyramid Networks for Object Detection.. In *CVPR*, Vol. 1. 3.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3431–3440.

David G Lowe. 1999. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, Vol. 2. Ieee, 1150–1157.

Frank Moosmann, Bill Triggs, and Frederic Jurie. 2007. Fast discriminative visual codebooks using randomized clustering forests. In *Advances in neural information processing systems*. 985–992.

Daniel Onoro-Rubio and Roberto J López-Sastre. 2016. Towards perspective-free object counting with deep learning. In *European Conference on Computer Vision*. Springer, 615–629.

Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. 2015. Cross-scene crowd counting via deep convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 833–841.

Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. 2016. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 589–597.

Zhaoxiang Zhang and Min Li. 2012. Crowd density estimation based on statistical analysis of local intra-crowd motions for public area surveillance. *Optical Engineering* 51, 4 (2012), 047204.