Asian hornets nest's detection on drone acquired FLIR and color images using deep learning methods

Tooba Shams
Univ. Bordeaux/Bees for Life https://www.beesforlife.fr/ contact@beesforlife.fr Email: toobashams8@gmail.com

Pascal Desbarats
Univ. Bordeaux, LaBRI UMR5800 (Univ. Bordeaux/CNRS/Bordeaux INP) 351, crs de la Libération, 33405 Talence cedex, France
Email: pascal.desbarats@labri.fr

Presented on the 2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA 2020)
Paris, France 9 – 12 November 2020

***Abstract*—Asian hornets are considered a pest because of their dangerousness and their impact on the ecosystem. Detecting nests of this species is a difficult task, as they are found in the trees, hidden in the leaves. Our goal is to carry out this detection from images acquired by a drone. We propose in this work a new method, based on the advantages of visible spectrum and FLIR images. We compare two models of state-of-the-art neural networks (YOLO and Mask-RCNN) for this task. The results are presented from the two separate image sets, then by combining the network responses. To do this, a third dataset (for ensemble model) was built by simulating a FLIR acquisition simultaneous with the acquisition in the visible spectrum. Preliminary results show that the best strategy is to use Mask-RCNN on the ensemble model (detection rate of 93%). A discussion on the relevant information present in the images and on taking into account of this information by the networks is also proposed.**

***Keywords*—** Object detection, Drone image acquisition, FLIR and visible spectrum images, Deep Neural Networks

I. Introduction

Asian hornet, also known as the yellow-legged hornet (Vespa velutina) is a species of hornets with bright yellow legs. Typically, queens reach 30mm in body length, males to 24mm, and most workers average 20mm [1]. Asian hornets primarily feed on larger insects, honey from the honeycomb, and tree saps; therefore, can be a problem for controlled environments of such insects and bees. An individual Asian hornet can approximately kill forty honeybees. Moreover, these hornets are very aggressive when in contact with humans. Having a stinger 6 mm long, it can inject a large amount of potent venom. In the case of multiple stinging, the result can be fatal. Hence, it is important to control the existence of such hornets in sites that serve other purposes.

This paper aims to use Forward-Looking Infrared (FLIR) images, and visible spectrum images acquired from drones to evaluate Neural Networks to detect Asian Hornet nest, ultimately testing if the two types of images work better individually or in a combined way. The work is done in context with the following problematics:

Occlusion: The Asian hornet nests are found in rooftops, trees, and other locations where other objects e.g. leaves usually occlude them. The insects build these paper-like structures in areas that have plenty of shade and protection from the elements. [2]; **Camouflage:** The size of an Asian hornet nest is big enough to be detected easily; however, the color is very similar to that of trees and leaves, which creates a camouflage effect; **Geometry and Point of view:** Due to the drone acquisition of images, the point of view is responsible for the geometry of a nest, making the detection a challenge; **Shape:** The nests of these hornets often vary in shape.

Separate models are built to detect nests in FLIR and color images, and later on the two models are combined intelligently to improve the results. A preliminary test is carried out for the combined model,

however the future goal of this research is to use synchronized images captured by a dual camera.

II. STATE OF THE ART

A. Nests detection

Studies have been made over finding nests by using conventional methods. In the paper [3], the authors proposed a method to find the nests by tracking worker hornets. The tracking is done using radio- telemetry. However, the strategy incorporates attaching tags to the hornets, taking around 10 to 15 minutes each, and afterwards testing whether the hornets could fly. [4] proposes a design of a harmonic radar for the tracking of the Asian hornet. However, this paper aims to detect the nests without touching the hornets. The problem puts this research under the domain of object detection and localization. It has been observed that the use of deep learning for this task has not been explored vastly.

B. Object detection

Object detection deals with detecting instances of objects of a particular class in digital images. In the past two decades, object detection has gone through two periods: "traditional object detection" (e.g. The Viola Jones algorithm [5] and SIFT [6]) and "deep learning- based detection" [7].

Deep learning-based object detection has been taking over the traditional object detection methods. Deep learning is a learning method with multiple levels of complexity, where the idea is to learn from low-level image features such as edges and lines to high-level features such as textures [8]. The representation on each level is called a feature map. Deep Neural Networks have been outstanding in finding significant structures in high dimensional data; hence, they are being used for object detection, specifically in the form of Convolutional Neural Networks CNNs. Machine learning techniques such as support vector machines (SVMs) [9] have gained popularity over time, as they can automatically learn the high-dimensional features of an image. Deep learning methods, together with the capacity of Graphics Processing Units (GPU), have proven to increase the framework efficiency [10]. Deep CNNs (DCNNs) are mostly used since they won the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) in 2012 by a model named AlexNet [11].

You Only Look Once (YOLO) is considered the state of the art CNN due to its real-time object detection capability. The authors propose to deal with object detection as a regression problem to the spatially separated bounding boxes and associated class probabilities [12]. YOLO performs object detection using a single neural network hence, making the training simple. Finally, the YOLOv2 [13] and YOLOv3 [14] improve the accuracy of detection and high-resolution classification, where YOLOv3 is just a slight increment in the performance of YOLOv2.

Region Convolutional Neural Networks (R-CNNs) have also been state-of-the-art, and after the advancements in R-CNNs, the object detection problem can be solved in real-time [15]. Among these advancements, Faster R-CNN [16] and Mask R-CNN [17] stay on top of the line due to their fast detection and training, where, Mask R-CNN adds a mask prediction branch, in parallel to the Faster R- CNN. For the above-stated reasons, all the experiments are carried

- **Log Average miss rate (LAMR):** The missed rate (MR) corresponds to the proportion of missed predictions over the total number of objects present in the ground-truth. The LAMR is the average of the missed rate on the logarithmic scale.
- out using YOLO and Mask R-CNN.

C. Ensemble Learning

$$MR = \frac{FN}{TP + FN} \quad (3)$$

Ensemble learning corresponds to strategically combining multiple models to solve a problem and is primarily used to improve the classification, prediction, or localization, depending on the nature of the problem [18]. In this paper, two networks are independently trained and combined to create an ensemble model. Several methods have been proposed for ensemble learning. [19] proposed to use a parallel network to detect Parkinson's disease. Similarly, Siamese Networks have also been used frequently ever since their break- through [20]. [21] proposes a score level fusion for multi-model bio- metrics recognition, and can be adjusted according to the problem into consideration.

III. EVALUATION FRAMEWORK

A. Evaluation Metric

The problem statement consists of object detection, localization, and segmentation. These conditions demand a metric that evaluates the classification and bounding boxes. [22] provides a framework for the evaluation of the chosen models using the Mean Average Precision (mAP) metric. While using Mask R-CNN, the predicted mask is inside a bounding box and only serves as a segmentation tool, hence ignored during the evaluation. FLIR model, color model, and the ensemble model are all evaluated separately. Another parameter called the Log Average Miss Rate (LAMR) is used to estimate missed predictions.

1) Terminologies:

- **Confusion Matrix:** Figure 1 shows the confusion matrix and its terminologies. A True Positive value refers to the detection of a nest present in an image. A False Positive detection means that a nest was not present but detected. A False Negative corresponds to missed detection. Finally, a True Negative implies that a nest was not present and not detected either.
- **Intersection over union (IoU):** IoU measures the overlap between two boxes and is used to measure the overlapping between the predicted bounding box and the ground truth bounding box.
- **Precision-Recall Curve:** The values of precision and recall range between 0 and 1 as the number of predictions progresses. A Precision-Recall curve plots the precision values against the recall values.

The metric takes the bounding box locations of ground-truth and predicted values as an input. For each detected nest, the IoU is calculated between the ground-truth and predicted bounding boxes. If the IoU is greater than a threshold, the detection is considered as a TP, and used to calculate the precision and recall values. Average precision computes the average precision value for recall value over 0 to 1, by calculating the area under the Precision-Recall curve. Finally, the mAP is the average AP over each class; however, in a binary classification, mAP is the same as AP.

B. Synthetic image generation

The FLIR and color images were taken at different times and were not synchronized. Therefore, FLIR images are synthesized using color images by performing color mapping and adding noise, to test the Ensemble Model. In image processing, color mapping refers to transforming the colors of an image from one domain to another and is done by using mapping functions. OpenCV color mapping function with the PLASMA palette is used to generate the corresponding FLIR images. According to the authors of [23], noise must always be considered in evaluating a thermal imaging system's performance. The IR sensors and the interference of the processing circuit are the primary sources of noise in FLIR images. The interference appears as thermal noise in the images and has an approximately Gaussian distribution; hence, a Gaussian noise with empirically chosen 0 mean and 0.5 standard deviation is used to model the thermal noise. The generated images do not carry FLIR properties, but serve the preliminary test purpose.

Fig. 1. Confusion Matrix

- **Precision:** This corresponds to the proportion of correct predic- tions over the total predictions.

IV. DATA

The dataset gathered consists of two parts, FLIR images and visible spectrum images. Both the sets are divided into three categories, including **Positive Images** (with the presence of a nest), **Negative Images** (without the presence of a nest), and **Suspicious Images** (Scenario where a nest might be present).

The images for the infrared dataset are taken with a FLIR Vue Pro R 640 13mm camera, having an image resolution of 640x512 pixels. This drone thermal camera produces radiometric JPEG im- ages, having temperature data embedded in each pixel. 1238 training images and 206 validation images are used to train the models. Fig. 2 and 3 represent FLIR dataset with positive and negative images. FLIR images being heat maps, are expected to detect occluded nests as well.

The images for the Color dataset are taken with a DJI FC6310S

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

camera, where most of the JPEG images have a resolution of 5472x3648 pixels. 2640 training images and 526 validation images

- **Recall (Sensitivity):** This corresponds to the proportion of correct predictions over the total number of objects present in the ground-truth.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

are used to train the models, due to the high complexity and number of available images. Fig.4 and 5 represent the visible spectrum dataset with positive and negative images respectively.

Only Positive and Negative images are used to obtain quantitative results.

Fig. 2. Representation of positive images of the FLIR Dataset

Fig. 3. Representation of negative images of the FLIR Dataset

Fig. 4. Representation of positive images of the color Dataset

Fig. 5. Representation of negative images of the color Dataset

A. Preparing the dataset

1) YOLO: You Only Look Once: The model is trained over the class ”nest”. The data annotations are made using positive images carefully selected from the dataset. For each image, a separate .xml file is generated using a basic code written in python. Each .xml file has the same name as the image file and contains the name of the image file, label, and coordinates of the location of a nest.

2) Mask R-CNN: A Mask R-CNN trains using training and validation datasets. The data annotations are in the form of JSON files, one each for the two datasets. The annotation file for the training dataset includes objects equal to the number of images in the dataset. Each object holds the annotation of one particular image.

B. Network selection

Terminologies

- **Optimization function:** Function used for minimization dur- ing training, such as Stochastic Gradient Descent (SGD) and momentum-SGD. This is also termed as loss function.
- **Epoch:** One iteration over the entire training data.
- **Batch size:** Number of images used to update the loss function.
- **Step:** An epoch contains several steps, each step uses one batch of images to update the loss function.
- **Checkpoint:** Steps after which a model is saved.

Table I summarizes the training of the two networks. The training is performed on the GTX1080 GPU, hence can vary while using other processors.

TABLE I
TRAINING INFORMATION

V. INDIVIDUAL NETWORKS

This section provides the implementation and comparison of two state of the art neural networks, to obtain the better performing network. Finally, this network is used to create individual models as well as the ensemble model. Fig. 6 shows the block diagram of the work.

Fig. 6. Required Network

1) YOLO: You Only Look Once: YOLO divides the entire image into 13x13 cells, where each cell is responsible for predicting 5 bounding boxes describing the object enclosed. Moreover, for each bounding box, YOLO also outputs a confidence score to define the class's certainty. In the end, the model provides a probability distribution over the classes on which the network has been trained. As this is preliminary work YOLO implemented in Tensorflow is used [1]. A validation set along with adaptive learning is added. Without a validation method, the model is prone to overfitting. In thiscase, the validation is done after every 10 steps. The learning rate is a critical parameter and a model greatly depends on it. This is a measure of the speed at which the neural network learns the training data, and is hard to tune. An adaptive learning technique is used to automatically tune the learning rate using the method described in [24]. Whenever the validation error plateaus, the learning rate is decreased to have more detailed learning.

[1]https://www.tensorflow.org/

2) Mask R-CNN: The notion behind a Mask R-CNN is to take an input image and output the bounding boxes' locations and their corresponding labels, which can be broken down into four steps.
- Region proposals.

$$\text{Class} = \text{BG}, \quad score < threshold$$
$$\text{Class} = \text{Nest}, \quad score \geq threshold \tag{5}$$

- Computation of convolutional features.
- Binary decision based on an SVM.
- Pixel level segmentation

Unlike YOLO, RCNN is a Support Vector Machine(SVM) based classifier modified to perform object detection. The model trains over 2 classes, "nest" and "background(BG)". The backbone of the Mask-RCNN used in this work is the Resnet101 [25] implemented in TensorFlow, which is a 101 layers deep convolutional neural network. The annotation files along with the images, contain the mask information. The network computes the bounding boxes from the input masks. Each image is resized to 1024x1024, to support multiple images per batch. A zero padding is added to the images to keep the aspect ratio preserved. The network only stores the mask's pixels

A drawback of Equation (4) is that if the color model is chosen to improve the FLIR model's predictions, and a nest is missed in a color image, the equation will force to remove the correct detection in the FLIR image as well. This can be explained through the following example. In a scenario where $\alpha = 0.2$ and threshold $= 0.5$, and the color model removes false predictions from the FLIR model. Assuming that the FLIR model makes a correct prediction and a false prediction with a score of 0.99 and 0.77 respectively, where 0.99 corresponds to a nest, and the color model misses the nest. The nest with a score of 0.99 will have a final score of 0.198 using Equation (4) and hence will be discarded by Equation (5). This demands to impose a condition on the system to apply Equation (4) only on nests with a score lower than a threshold resulting in Equation (6)

inside the bounding boxes to save memory and increase training speed. The mask can be resized to a smaller size, 56x56, in this

$$\text{if} \quad score = \alpha S_F + (1-\alpha) S_V, \quad S_F < Th_S$$

(6)

case, for further speed. The region proposal network uses sliding windows to calculate anchors. For an input image of 1024x1024, the feature map of the first layer is 256x256. This results in 200,000 anchors containing significant overlaps, and can be configured using the stride, 2 in this case. Positive, negative, and neutral anchors are generated; however, neutral anchors are not used for training.

An instance of the model is created in inference mode for predic- tions. For a test image, the region proposal network runs a binary classifier on the anchors over the image to give an object score. The classification network classifies whether an anchor contains an object or not and assigns a probability to the positive anchors. Finally, the mask is generated from the bounding box.

TABLE II

$score = S_F,$ otherwise

Where Th_S is a score threshold used to apply the fusion. Equation (5) and (6) are finally used to create the Ensemble Model.

VII. RESULTS & DISCUSSION

Table III summarizes the testing results of all the models using Equation (5) and (6) with the empirically chosen parameter shown in Equation (7) and (8). GT equals Ground Truth nest instances. The value of Th_S is set to 0.98 as most of the TP nests have a confidence of 0.99. As the Mask R-CNN based FLIR model produces a higher number of FPs, only those nests that have a high score in the color images are kept. A nest is detected if the final score is greater than 0.5.

NETWORK SELECTION

	YOLO FLIR	**YOLO Color**		

Based on the results in Table II, Mask R-CNN is used to create the Ensemble Model as it significantly outperformed YOLO.

VI. ENSEMBLE MODEL

This section presents the design of the ensemble model using individual Mask R-CNN based FLIR and color models.

A. Score level fusion

The ensemble model can be visualized as an analogy to multimodel biometric recognition, where the same person is recognized based on various inputs such as fingerprints, facial features, or iris recognition. A traditional approach in biometric identification is to combine the scores of various biometric features, corresponding to score level fusion which results in a single scalar value, as mentioned in [26]. The confidence scores output by multiple inputs are consolidated to render a decision.

A. YOLO

$$score = 0.3S_F + 0.7S_V, \quad S_F < 0.98$$
$$score = S_F, \quad \text{otherwise}$$

$$\text{Class} = \text{BG}, \quad score < 0.5$$
$$\text{Class} = \text{Nest}, \quad score \geq 0.5$$

TABLE III RESULTS

(7)

(8)

Each individual model outputs a bounding box, a mask, and aconfidence score for a detected nest. The two scores are combine to give a final score using a weighted average, shown in Equation (4).

$$score = \alpha S_F + (1-\alpha) S_V \quad (4)$$

Where α ranges between 0 and 1, S_F and S_V are the confidence scores predicted from FLIR and visible spectrum images, respec- tively. An increase in α increases the significance of S_F in the final score, and vice versa. Finally, the resulting score is compared with a threshold, and the nests below the threshold are considered background (BG), as shown in Equation (5).

Due to its literature review, YOLO was the first choice for this object detection task. However, as evident from Table III, YOLO did not perform as required. YOLO imposes spatial constraints on bounding box predictions as each cell predicts five bounding boxes with one class each, and this process takes place once. This limits the number of object detections. Moreover, the geometry and shape of the nest can vary from image to image. A nest can be small in size in some images. YOLO, however, can struggle with precisely locating small nests. Finally, since YOLO has many downsampling layers, a small-sized unclear object can be hard to detect, implying localization inaccuracies. Hence, YOLO failed to perform in thistask.

B. Mask R-CNN

The Mask R-CNN based models substantially outperformed YOLO with a higher mAP and a lower LAMR. Since Mask R- CNN efficiently generalizes different poses and geometries, it can detect nests in more scenarios than YOLO. Moreover, as each region proposed by the Region Proposal Network passes through an SVM based classifier, the resulting regions are more robust. Furthermore, since the Mask R-CNN runs a linear regression to tighten the bounding boxes, it reduces the localization error, improving mAP value. Table III shows that the the detection of a nest using FLIR images slightly outperforms detection based on color images in terms of the mAP value and number of True Positives, because the FLIR images can handle occlusion and camouflage better than the color images. However, with a higher number of TPs, the FLIR based model also has a higher number of FPs. Since a FLIR image has thermal information encoded in the pixels, heat from other sources can also correspond to a nest, causing false detections. For example, in Figure 2, the multiple bright spots can be misinterpreted as anest. Hence, the ensemble model uses color images to remove false detections, thus providing better results.

C. Ensemble Model

The Ensemble Model combines the Mask R-CNN based FLIR and color models. The value of α needs to be set such that the two models complement each other. As evident from Table III, the ensemble model significantly reduces the number of False Positives. Moreover, a slight increase in the final mAP value is seen. As the FLIR model is capable of detecting more nests but also generates a higher number of False Positives, the color model is used to remove the FPs. Moreover, if a nest is missed by the FLIR model and is detected by the color model, it is considered as a TP. The ensemble model results outperform both the individual models in terms of FPs, mAP, and LAMR. The Ensemble model is tested on synthetic FLIR images along with their corresponding color images.

VIII. CONCLUSION

Visible spectrum images only provide information that is on the surface. If a nest is occluded, it does not appear in the images and is likely to be missed. On the other hand, color images have more texture and information, resulting in more robust learning features. Forward-Looking Infrared (FLIR) images have a lower amount of learning information as they encode the heat information in each pixel. The heat emitted from a nest and a streetlight both can appear very similar in a FLIR image, therefore, be treated as the same object. Contrary to this, occluded nests are easily visible, improving the chances of detections. Regarding precision, the results were promising. Mask R-CNN based models significantly outperformed YOLO based models. Even the independent data gave good results, it is better to use the ensemble model. For this reason synchronized FLIR color data will be acquired. This research has provided a way of detecting the presence of Asian Hornet nests effectively using DNNs. For the future research, evaluation of live streaming of images from the drones to the computer can be done, or on video acquisitions.

Other fusion methods can be explored. Finally, a reinforcement learning technique can be explored to improve models over time.

ACKNOWLEDGMENT

We would like to thank Bees For Life, for collecting all the data and making it available for this work, and European Union and the Erasmus Mundus Association along with Bees For Life for funding this research.

REFERENCES

[1] de Haro, Luc, et al. "Medical consequences of the Asian black hor- net (Vespa velutina) invasion in Southwestern France." Toxicon 55.2-3 (2010): 650-652.

[2] Perrard, Adrien, et al. "Observations on the colony activity of the Asian hornet Vespa velutina Lepeletier 1836 (Hymenoptera: Vespidae: Vespinae) in France." Annales de la Société entomologique de France. Vol. 45. No. 1. Taylor Francis Group, 2009.

[3] Kennedy, Peter J., et al. "Searching for nests of the invasive Asian hornet (Vespa velutina) using radio-telemetry." Communications biology 1.1 (2018): 1-8.

[4] Milanesio, Daniele, et al. ”Design of an harmonic radar for the tracking of the A sian yellow-legged hornet.” Ecology and evolution 6.7 (2016): 2170-2178.
[5] Viola, Paul, and Michael Jones. ”Robust real-time object detection.” International journal of computer vision 4.34-47 (2001): 4.
[6] Piccinini, Paolo, Andrea Prati, and Rita Cucchiara. ”Real-time object detection and localization with SIFT-based clustering.” Image and Vision Computing 30.8 (2012): 573-587.
[7] Zou, Zhengxia, et al. ”Object detection in 20 years: A survey.” arXiv preprint arXiv:1905.05055 (2019).
[8] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. ”Deep learning.” nature 521.7553 (2015): 436-444.
[9] Schölkopf, Bernhard, and Alex Smola. ”Support vector machines.” Wiley StatsRef: Statistics Reference Online (2014).
[10] Pietrow, Dymitr and Jan Matuszewski. “Objects detection and recog- nition system using artificial neural networks and drones.” 2017 Signal Processing Symposium (SPSympo) (2017): 1-5.
[11] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. ”Imagenet classification with deep convolutional neural networks.” Advances in neural information processing systems. 2012.
[12] Redmon, Joseph, et al. ”You only look once: Unified, real-time object detection.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[13] Redmon, Joseph, and Ali Farhadi. ”YOLO9000: better, faster, stronger.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
[14] Redmon, Joseph, and Ali Farhadi. ”Yolov3: An incremental improve- ment.” arXiv preprint arXiv:1804.02767 (2018).
[15] Girshick, Ross, et al. ”Rich feature hierarchies for accurate object de- tection and semantic segmentation.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
[16] Ren, Shaoqing, et al. ”Faster r-cnn: Towards real-time object detection with region proposal networks.” Advances in neural information process- ing systems. 2015.
[17] He, Kaiming, et al. ”Mask r-cnn.” Proceedings of the IEEE international conference on computer vision. 2017.
[18] Polikar, R. ”Ensemble Learning In Ensemble Machine Learning: Meth- ods and Applications. Zhang, C.; Ma, Y., Eds.” (2012): 1-34.
[19] Åström, Freddie, and Rasit Koker. ”A parallel neural network approach to prediction of Parkinson’s Disease.” Expert systems with applications 38.10 (2011): 12470-12474.
[20] Bromley, Jane, et al. ”Signature verification using a” siamese” time delay neural network.” Advances in neural information processing systems. 1994.
[21] Aizi, Kamel, and Mohamed Ouslim. ”Score level fusion in multi- biometric identification based on zones of interest.” Journal of King Saud University-Computer and Information Sciences (2019).
[22] Dollar, Piotr, et al. ”Pedestrian detection: An evaluation of the state of the art.” IEEE transactions on pattern analysis and machine intelligence 34.4 (2011): 743-761.
[23] Kennedy, Howard V.”Modeling noise in thermal imaging systems.” Infrared Imaging Systems: Design, Analysis, Modeling, and Testing IV. Vol. 1969. International Society for Optics and Photonics, 1993.
[24] Moreira, Miguel, and Emile Fiesler. Neural networks with adaptive learning rate and momentum terms. No. REP WORK. Idiap, 1995.
[25] He, Kaiming, et al. ”Deep residual learning for image recognition.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
[26] Daniel, David Marius, C˘as¸lariu Mihaela, and Terebes¸ Romulus. ”Com- bining feature extraction level and score level fusion in a multimodal biometric system.” 2014 11th International Symposium on Electronics and Telecommunications (ISETC). IEEE, 2014.