2020 17th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)
Mexico City, Mexico. November 11-13, 2020

Analysis of CNN Models to develop a New Appearance Model for Multiple-Object-Tracking

Francisco Javier Alvarez-Prieto
Visual Perception Lab
Tecnologico Nacional de Mexico/I.T.
Chihuahua, Chih, Mexico
fjalvarezp@itchihuahua.edu.mx

Mario I. Chacon-Murguia
Visual Perception Lab
Tecnologico Nacional de Mexico/I.T.
Chihuahua, Chih, Mexico
mchacon@ieee.org

Juan A. Ramirez-Quintana
Visual Perception Lab
Tenologico Nacional de Mexico/I.T
Chihuahua, Chih, Mexico
jaramirez@itchihuahua.edu.mx

***Abstract*— Multiple-object-tracking, MOT, is a paramount research area in computer vision tasks such as: intelligent surveillance systems, motion analysis, self-driving, etc. Different techniques have been considered for the solution of MOT, however, recent works are aimed to involve deep neural networks. Therefore, this paper presents an analysis of different convolutional neural networks (CNN) reported in the literature for MOT. The aim of this analysis is to determine which CNN models perform better when they are evaluated in the main challenges found in MOT. Then, based on the analysis results, a new robust appearance model generated from a Siamese CNN features is proposed. Findings indicate that CNN with fewer layers perform better than CNN with more layers, which may be to the fact that more trivial features allow a better discrimination among the objects of the same class.**

Keywords*— multiple-object-tracking, appearance model, convolutional neural networks.

I. Introduction

Video surveillance, sport analysis, event detection, are examples of visual applications that require the intervention of experts to continuously review and analyze recorded videos. This kind of human analysis conveys several disadvantages because of the reduced human attention capacity to perform a repetitive task during a long period of time. In some cases, current technology is used to replace human intervention in these types of activities. However, considering the level of complexity found in this activity more research is still needed. An important factor in the performance of this technology is to have a robust algorithm to achieve multiple-object-tracking (MOT). At the same time, the MOT algorithm relies on its performance in two components: a motion model to predict the position of the objects or targets in the next frames, and an appearance model (AM) to model the aspect of the objects during the video sequence and to establish the correct identity correlation of the different objects. The AM is of great importance, since several authors agree that it is crucial in the performance of a MOT algorithm [1]-[4]. Considering the previous information, this work presents the analysis of several convolutional neural networks (CNN) that may be employed to develop an AM. The analysis, among other issues, considers the ability of the CNN to adapt the AM in different challenges commonly present during MOT.

One contribution of this work is the analysis of several CNN models used to yield AMs, considering their robustness to discriminate objects during the tracking process in several motion challenges; and to propose a new AM based on a Siamese CNN. The experiments show that the Siamese CNN has better discrimination of objects in comparison with the traditional CNN analyzed.

II. Literature Review

MOT is a highly active research area that involves a broad work on the literature. Recently, several AM based on Deep neural networks (DNN) have been proposed in the literature. For example, Hong *et al,* [5] combine a pretrained CNN with a support vector machine to learn the appearance of the objects by obtaining discriminants from the feature maps. According to their results, the precision increased in the location of the objects with regard the best ten methods in the VT *Benchmark*. Zhu *et al.* [6] trained a CNN with spatial attention able to relate coincident patterns of specific sections of two input images, which allow obtaining better performances in terms of metrics to preserve the object identity in comparison with other methods in MOT *Benchmark*. Yoon and Bae [7] and Kieritz *et al.* [8] propose a deep learning method to generate a sequential discriminative AM of each object. This model can track multiple objects online. In [9] Kim *et al.* describe a new Bilinear Long Short-Term Memory (LSTM) model network. This model allows the sequential long-term learning of objects appearance. Sadeghian *et al.* [10] combine three recurrent neural networks (RNN) to compute, appearance, motion and social interaction, based on similarity. This combination allows correcting errors in data association and to recover objects identity during occlusions. Fang *et al.* [11] employ people reidentification techniques and localization information in an autoregressive RNN (RAN). According to their results, the RAN presented better robustness during occlusions in crowd scenes compared to other methods in MOT *Benchmark*. In other work [12] Chu *et al.* developed an *end-to-end* model. In this model, feature extraction, affinity measure and the multidimensional assignment are achieved with only one deep network termed FAMNet. This network is jointly optimized to generate the trajectories of the targets. Considering the results reported in that work, the network acquires the capacity to learn in an integral form the features of high-level affinity. The work of Leal *et al.* [13] presents a Siamese CNN to relate detected targets between adjacent frames using the output of the Siamese network as a similarity metric. The results of this work indicate a greater accuracy in data association considering other methods in MOT *Benchmark*. It is important to note that our proposed Siamese Network is different from Leal *et al.* [13]. Our proposed network has a different architecture. It was defined first by analysis of the deep of other CNN, and its parameters were adjusted considering a discrimination level.

Regarding MOT evaluation methods, the most used in the literature is CLEAR MOT, which considers the metrics MOTP (Multi-Object Tracking Precision) and MOTA (Multi-Object Tracking Accuracy). However, in Section E of this work, a new metric called discrimination level is introduced. This new metric evaluates the performance of AM and can be

978-1-7281-8987-1/20/$31.00 ©2020 IEEE

interpreted as the easiness to distinguish an object of interest from the others.

III. METODOLOGY

This section describes the methodology to achieve the evaluation of the performance of the CNN models analyzed and presents the proposed appearance model.

A. Test Scenarios

In order to evaluate the performance of the distinct CNN, there were considered six types of test scenarios which include different challenges or categories commonly included in MOT: changes of appearance, scale, illumination, similar appearance, noise and occlusions [14]. Each category is described below.

- Appearance change (AC): These scenarios include objects that change their physical aspects, e.g. objects seen from different angles.
- Scale variations (SV): This category includes objects captured by moving cameras, where in the first frames the objects are small, but as the camara approaches the object size increases.
- Illumination change (IC): This situation was simulated by gradually adjusting the brightness of the objects.
- Similar appearance (SA): The set involves objects with similar physical aspects, e.g. people wearing clothes of the same color.
- Image noise (IN): The noise was simulated by adding different levels of Gaussian and salt and pepper noise to the video.
- Occlusion (OCL): This set includes sequences of frames where the objects are completely visible at the beginning of the sequence, and then they are gradually occluded by other objects until they are completed occluded.

Each set of test scenarios has five targets captured during 15 frames, taken from the database MOT16 [15]. This database is ideal to test surveillance systems because the objects of interest are only persons. Fig. 1 illustrates an example of each category of the scenarios.

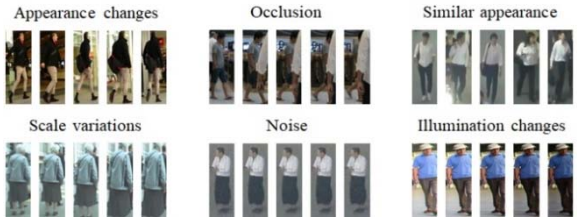

Fig. 1. An example of objects of each one of the 6 categories.

B. Selection of the CNN models

The CNN considered in this work are: Alexnet, ResNet-18, VGG-16, Inception-ResNet-v2, DenseNet-201 and NASNet-Large. These models were selected to cover the broad spectrum of CNN strategies proposed in the literature. Some of the most popular CNN are illustrated in Fig. 2. The positions of the CNN in the plot correspond to the relation between their accuracy and their processing time to produce an output. The CNN selected for this works are indicated in orange diamonds.

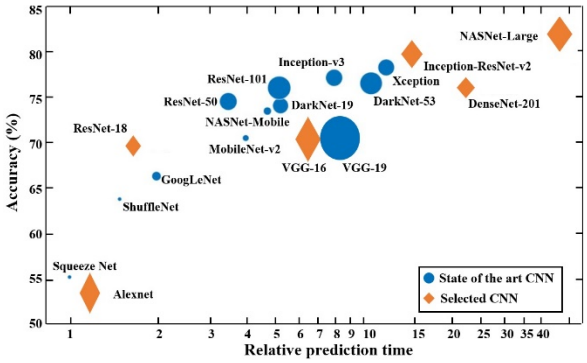

Fig. 2. Accuracy vs. Processing time of the state of the art CNN used in classification. The selected CNN are indicated in orange diamonds.

A CNN architecture is mainly composed of two stages: feature extraction and classification. The first stage includes convolutional and pooling layers. The second stage is distinguished by incorporating at least one fully connected layer to perform classification.

Considering FM_i^l like the *ith* feature map of dimension $h^l \times \omega^l$ in the layer l, where h^l and ω^l are the height and width, respectively. The output of the different layers is computed as follows.

Convolutional layer: FM_i^{l-1} is convolved with several filters $w_{i,j}^l$ of dimension $m^l \times m^l$ in the layer l. The result is evaluated with a non-linear activation function $f(\cdot)$, commonly a rectified linear unit (ReLU). Finally, a *jth* feature map FM_j^l of dimension $(h^{l-1} - m^l + 1) \times (\omega^{l-1} - m^l + 1)$ is obtained with

$$FM_j^l = f\left(\sum_{i=1}^{k^l} FM_i^{l-1} * w_{ij}^l + b_j^l\right) \quad (1)$$

where b_j^l is the bias factor and k^l is the number of filters used in the layer l, such that FM_i^{l-1} corresponds to the input image O when $l=1$.

Pooling layer: This layer reduces the dimension of FM_j^{l-1} by performing a subsampling with a scale factor of r. The max-pooling technique is the most used subsampling operation. Thus, a new feature map FM_j^l is computed by

$$FM_j^l = \max\_p\left(FM_j^{l-1}\right) \quad (2)$$

here, $max\_p(\cdot)$ consists on taking the maximum value in each region $r \times r$ of FM_j^{l-1}.

Fully connected layer: In the case of the first fully connected layer, it is necessary to concatenate all features maps of the previous layer, that is $DF^{l-1} = \left[FM_{j=1}^{l-1}, \ldots, FM_{k^{l-1}}^{l-1}\right]$ to obtain a vector representation. Then, DF^{l-1} is multiplied with the weight matrix W^l of the neurons in the fully connected layer. The result of the multiplication is passed through a non-linear activation function, typically *softmax*. Finally, the new output vector DF^l is computed by

$$DF^l = f\left(W^l DF^{l-1} + b^l\right) \quad (3)$$

If there are more fully connected layers, the output of those layers is obtained by (3) but considering the vector DF^l as the input to the new layer, and the weights matrix and bias of the next fully connected layer.

C. Feature Extraction

To evaluate the performance of the selected CNN features in MOT an experiment was designed. The first stage of the experiment consists on the feature vector extraction of the objects of the six test categories. The second stage evaluates the features of the objects to determine their similarity. The third stage employs the similarity obtained in the second stage to determine the level of discrimination and robustness of the CNN under analysis.

In this work the feature vector, termed descriptor, is obtained by the concatenation of all the feature maps of the layer *L-1*

$$DF^{L-1} = \left[FM_{j=1}^{L-1}, ..., FM_{k^{l-1}}^{L-1} \right] \tag{4}$$

where L indicates the first fully connected layer of the CNN model, Fig. 3. For simplicity and considering that in this experiment each descriptor is obtained in the layer *L-1*, this notation is omitted and DF_i^j stands for the descriptor of the *ith* object in the *jth* frame.

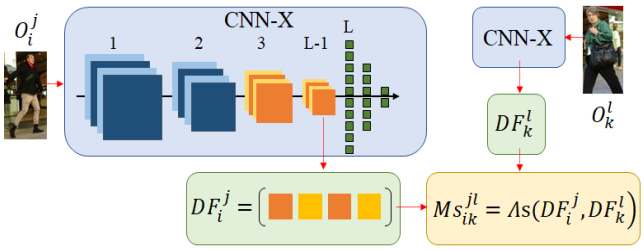

Fig. 3.Obtention of the similarity metric Ms_{ik}^{jl} between objects.

D. Similarity Function

The descriptor of each object is evaluated by a similarity function $\Lambda s(\cdot)$, to determine the level of similarity *Ms*, used to decide if the target under analysis is the same target or not. Two $\Lambda s(\cdot)$ are employed in this work. The cross-correlation coefficient ρ, and the Euclidean distance *Ed*. Before the computation of *Ed,* the descriptors *DF* are normalized to unitary vectors represented by $\overline{DF}$.

$$\rho = \frac{\sum_{n=0}^{N-1} \left(DF_i^j[n] \right) \cdot \left(DF_k^l[n] \right)}{\sqrt{\sum_{n=0}^{N-1} \left(DF_i^j[n] \right)^2 \cdot \sum_{n=0}^{N-1} \left(DF_k^l[n] \right)^2}} \tag{5}$$

$$Ed = \sqrt{\sum_{n=0}^{N-1} \left(\overline{DF_i^j}[n] - \overline{DF_k^l}[n] \right)^2} \tag{6}$$

where DF_k^l is the descriptor of the *kth* object in the *lth* frame. Therefore, Ms_{ik}^{jl} represents the similarity, Fig. 3, between the target O_i^j with respect the target O_k^l. Thus, a value of Ms_{ik}^{jl} close to 1 obtained with the correlation ρ or a value close to 0 using *Ed* indicates that the targets are more similar.

Fig. 4 shows an example of the similarity metric for the appearance change scenario between the target 5 in the frame *j*, itself and the other four objects in the frame *j*+1 during 15 frames. Due to space restrictions, the similarity is only shown when the descriptors of CNN Alexnet and the Inception-ResNet-v2 are used. The example makes evident that the discrimination of the object is better when the descriptors of the Alexnet are used. It can be observed that the margin of separation is consistent between the object of interest and the other objects. This result is not observed when the descriptors of Inception-ResNet-v2 are employed, because the separation margin is smaller compared to the Alexnet results. The worst case in the Inception-ResNet-v2 is in the frame 8, indicated with a red arrow. It is when another object is confused with the object of interest. Based on these observations, a discrimination level *Dl* that allows to resume the information of Fig.4 was generated.

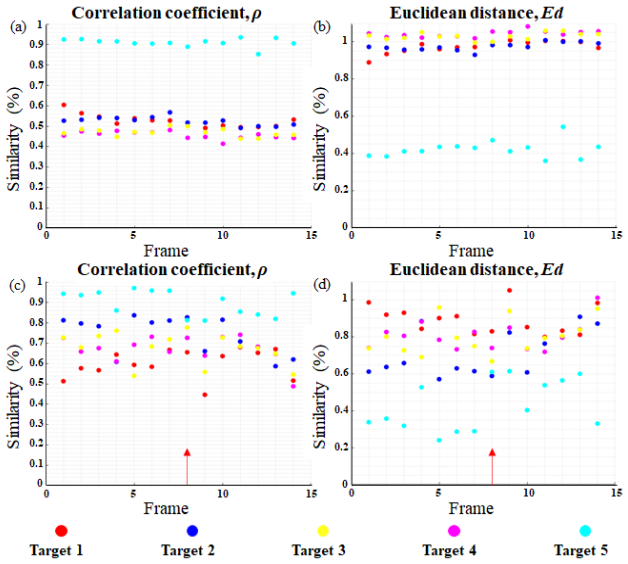

Fig. 4. Similarity measure using the descriptors of CNN Alexnet (a)(b) , and Inception-ResNet-v2 (c)(d).

E. Discrimination Level Dl

The discrimination level *Dl* is computed as follows. Assume the similarity metric between the object of interest with itself

$$MsOI_i^j = \Lambda s \left(DF_i^j, DF_i^{j+1} \right) \tag{7}$$

and the similarity metric of the object of interest and the other objects as

$$MsOP_i^j = \Lambda s \left(DF_i^j, DF_{\neq i}^{j+1} \right) \tag{8}$$

Note that only the similarity metric is computed on consecutive frames (*j* and *j*+*1*). Therefore, the appearance model is an updated appearance model because it is updated after processing each video frame. Thus, the separation margin in the frame *j* for the object *i*, Se_i^j, is given by (9) when ρ is employed, and for (10) when *Ed* is used.

$$Se_i^j = MsOI_i^j - \max \left(MsOP_i^j \right) \tag{9}$$

$$Se_i^j = \min \left(MsOP_i^j \right) - MsOI_i^j \tag{10}$$

Now, since each test scenario contains five objects captured during 15 frames, the *Dl* of each CNN for a specific challenge is computed by the average of the separation margin considering the 15 frames for the five objects (11).

$$Dl = \frac{1}{5} \sum_{i=1}^{5} \frac{1}{14} \sum_{j=1}^{14} Se_i^j \tag{11}$$

Dl can be interpreted as the easiness to distinguish an object of interest from the other objects in the test, in a specific

MOT challenge. So that, as higher the value of *Dl* the challenge will be overcome. However, it is important to note that there is no direct comparison between the results using ρ and *Ed* as the similarity function.

Table I shows the values of *Dl* obtained for each CNN in each one of the six test scenarios when ρ is employed, and Table II for the case of the *Ed*. This information is also illustrated in Fig. 5 and Fig 6 respectively.

TABLE I. *DL* OF THE CNN USING ρ AND THE UPDATED AM APPROACH.

	IC	SA	AC	SV	OCL	IN
Alexnet	0.62	0.36	0.46	0.39	0.45	0.33
VGG-16	**0.72**	**0.40**	**0.49**	**0.41**	**0.46**	**0.36**
ResNet-18	0.17	0.13	0.13	0.14	0.18	0.11
DenseNet-201	0.37	0.21	0.26	0.26	0.35	0.24
Inception-ResNet-v2	0.16	0.09	0.12	0.09	0.16	0.11
NASNet-Large	0.34	0.20	0.23	0.20	0.30	0.26

TABLE II. *DL* OF THE CNN USING *ED* AND THE UPDATED AM APPROACH.

	IC	SA	AC	SV	OCL	IN
Alexnet	1.02	**0.53**	**0.60**	**0.46**	**0.50**	**0.43**
VGG-16	**1.06**	0.46	0.55	0.41	0.44	0.38
ResNet-18	0.51	0.28	0.30	0.26	0.32	0.21
DenseNet-201	0.74	0.33	0.42	0.35	0.43	0.35
Inception-ResNet-v2	0.49	0.22	0.28	0.20	0.29	0.24
NASNet-Large	0.71	0.31	0.38	0.28	0.37	0.37

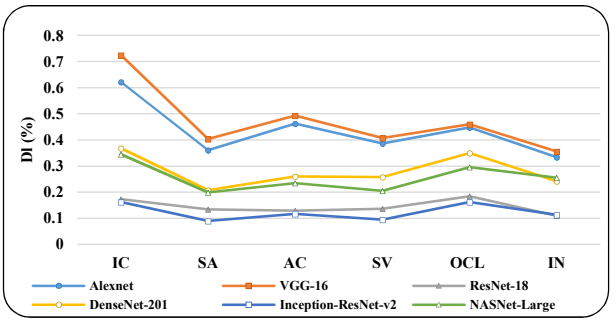

Fig. 5. *Dl* of the CNN using ρ and the updated AM approach.

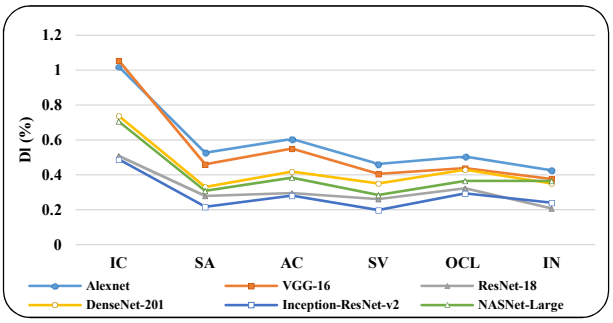

Fig. 6. *Dl* of the CNN using *Ed* and the updated AM approach.

At this point, the experiment only considered the similarity between objects in consecutive frames. In order to verify the robustness of the AM built from the descriptors of the CNN, a new experiment using a non-updated AM was achieved. That is, the AM is generated in the first frame and it is kept without change during the complete video sequence. In this experiment the similarity metric of the object of interest with itself, and the object of interest with the other objects in the test are computed by

$$MsOI_i^j = \Lambda s\left(DF_i^1, DF_i^{j+1}\right) \tag{12}$$

and

$$MsOP_i^j = \Lambda s\left(DF_i^1, DF_{\neq i}^{j+1}\right) \tag{13}$$

In this experiment, the similarity metric is computed between the objects in the frame 1 and the frames *j+1: j=1, 2,...,14* . This allows to test the robustness of the descriptors because the similarity metric is computed temporally with more distant objects, which may include drastic changes. The *Dl* in this experiment is also computed by (9)-(11). The *Dl* obtained for each CNN using ρ in this version of the experiment is shown in Table III and Fig. 7, and Table IV and Fig. 8 show the values of *Dl* employing *Ed*.

TABLE III. *DL* OF THE CNN USING ρ AND NON UPDATED AM APPROACH.

	IC	SA	AC	SV	OCL	IN
Alexnet	0.59	**0.24**	**0.23**	**0.17**	**0.19**	**0.27**
VGG-16	**0.67**	0.25	0.23	0.13	0.16	0.23
ResNet-18	0.19	0.13	0.07	0.07	0.08	0.04
DenseNet-201	0.35	0.13	0.14	0.14	0.15	0.16
Inception-ResNet-v2	0.24	0.07	0.09	0.03	0.05	0.07
NASNet-Large	0.31	0.10	0.11	0.08	0.13	0.16

TABLE IV. *DL* OF THE CNN USING *ED* AND NON UPDATED AM APPROACH.

	IC	SA	AC	SV	OCL	IN
Alexnet	0.86	**0.31**	**0.27**	**0.18**	**0.20**	**0.27**
VGG-16	**0.87**	0.27	0.23	0.12	0.14	0.20
ResNet-18	0.46	0.21	0.14	0.10	0.12	0.05
DenseNet-201	0.62	0.19	0.20	0.16	0.17	0.18
Inception-ResNet-v2	0.52	0.13	0.16	0.04	0.08	0.10
NASNet-Large	0.56	0.16	0.16	0.10	0.15	0.18

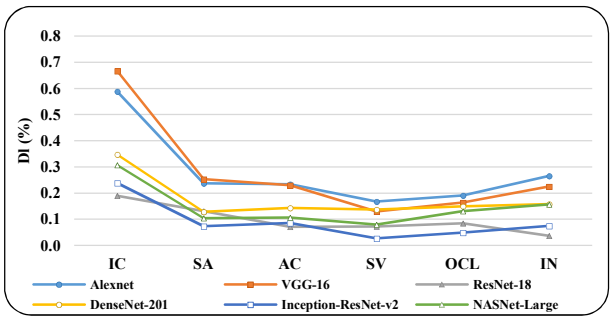

Fig. 7. *Dl* of the CNN using ρ and the non updated AM approach.

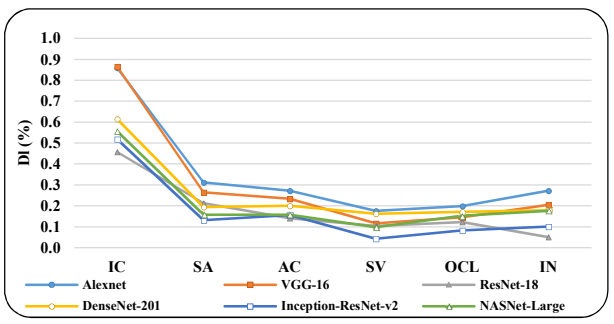

Fig. 8. *Dl* of the CNN using *Ed* and the non updated AM approach.

The results generated when an update AM is used indicate that Alexnet and VGG-16 report the best performances. That means that in these experiments Alexnet and VGG-16 are

more recommended solving the different MOT challenges. It is worth mentioning that Alexnet and VGG-16 are the least deep CNN considered in the experiment. This may suggest that as the deep of the CNN decreases its capacity to differentiate objects increases. This issue is different from other CNN applications like classification, where the deepest CNN performs the best.

The previous results can be explained by looking at the information encoded by the deepest layers of the CNN. As mentioned in [16], the information in the deepest layers is associated with the semantic content of the objects and as the depth of the CNN increases, the quality of semantic information improves. Semantic information is particularly useful to discriminate objects from different classes, but not when objects belong to the same class. Therefore, networks with less capacity to describe semantic content (those that are shallower) obtain better results to discriminate objects of the same class.

In the case when the AM is not updated the less deep CNN are still the best CNN. However, their performance is considerably reduced compared with an AM updated approach. This may lead to conclude that none of the CNN studied are robust enough to model drastic changes of object appearance.

F. Proposed Appearance Model

This section describes the proposed AM based on a Siamese CNN configuration. This configuration involves two identically CNN with the same parameters. The main difference between this neural architecture and a traditional CNN is that the Siamese network is fed with two images, that may be a positive pair (images of the same object) or a negative pair (images of different objects). Both images are propagated through the network to generate their respective descriptors, and then the similarity between them is computed. Thus, the output of the network is the similarity value. One advantage of this network configuration is that the similarity and dissimilarity concepts are directly related to the learning algorithm of the network.

The architecture of the Siamese network was defined by considering the previous experiments and the depth of the CNN analyzed. It was decided to design a network with a 50% reduction in the convolutional layers compared to CNN Alexnet, which is the least deep CNN considered in the experiment. Thus, the Siamese network has three convolutional layers (C_l), three pooling layers (P_l) and a fully connected layer (FC) represented by equations (1) to (3). Fig. 9 illustrates the proposed Siamese network, where the final block corresponds to the similarity function ρ defined in (5). The information of each layer is presented in Table V.

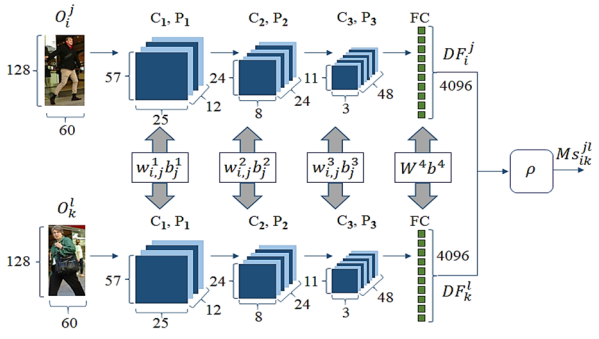

Fig. 9. Appearance model based on a Siamese CNN.

The training of the Siamese network was achieved by batches of size of 180 and 1500 iterations. The learning rate was 0.00001 and the cost function was the binary cross-entropy. The training images were from the database CUHK03 [17], which was designed for person reidentification task in views of 10 different angles. Equal number of positive and negative pairs were provided during the training. The images were normalized by the z-score normalization process. To produce image values with zero mean and unitary standard deviation.

TABLE V. INFORMATION OF THE PROPOSED SIAMESE CNN.

Layer type	Layer properties
Image input	128x64x3 with z-score normalization
Convolutional layer 1, C_1	12 filters 15x15x3 with stride [1 1], padding [0 0 0 0] and ReLU
Max pooling layer 1, P_1	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
Convolutional layer 2, C_2	24 filters 9x9x12 with stride [1 1], padding [0 0 0 0] and ReLU
Max pooling layer 2, P_2	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
Convolutional layer 3, C_3	48 filters 3x3x24 with stride [1 1], padding [0 0 0 0] and ReLU
Max pooling layer 3, P_3	2x2 max pooling with stride [2 2] and padding [0 0 0 0]
Fully connected layer, FC	4096 neurons with sigmoidal

To validate the performance of the proposed network, it was compared with the CNN VGG-16 because it obtained the best performance with the similarity function ρ in the experiment of *Dl*. Fig. 10 and 11 show the comparison considering the updated and non-updated AM approaches, respectively.

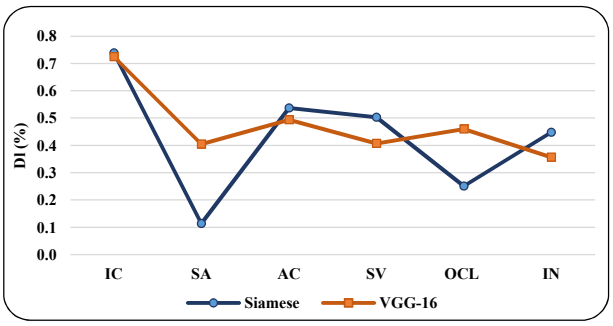

Fig. 10. Comparison between the Siamese and VGG-16 networks considering *Dl*, ρ, and the updated AM approach.

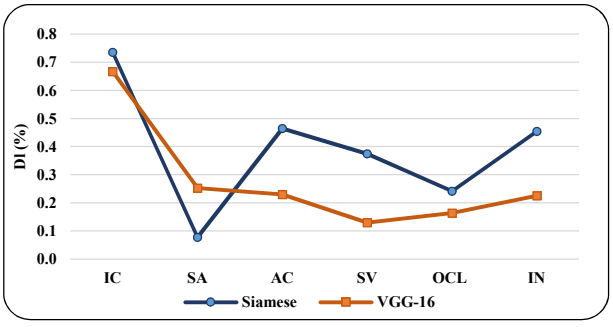

Fig. 11. Comparison between the Siamese and VGG-16 networks considering *Dl*, ρ, and the non-updated AM approach.

IV. Results

The results illustrated in Fig. 10 show that the proposed model outperform the results of the VGG-16 in the challenges AC, SV, and IN, and slightly better in IC. VGG-16 is better in the challenges SA and OCL. Nevertheless, it can be noticed in Fig. 11, that the proposed model is far better than VGG-16 AC, SV and IN, it is lightly better in IC and OCL, and it is only overcome in SA.

An important point is that the *Dl* in the Siamese network stays almost the same in both experiments. On the other hand, the *Dl* of VGG-16 drastically decays when the AM is not updated. This is an indication that the proposed network can model the appearance of objects even they are not in consecutive frames. Therefore, it is a more robust network to face drastic object changes. However, this advantage limits its capacity to distinguish different objects with similar appearance. These results, as it was mentioned in Section E, are related with the semantic content present in the layers of the CNN and the concepts of similarity and dissimilarity introduced during the training of the Siamese network.

V. Conclusion

This paper presented an analysis of different CNN used to generate an AM used in MOT. Based on this analysis a new Siamese network was proposed for MOT.

The experimental results indicate that CNN with fewer layers like the Alexnet and the VGG-16 can discriminate better objects in different MOT challenges. Therefore, deeper CNN employed in this work present lower capacity to discriminate objects in MOT.

Regarding the proposed network, the findings show that the Siamese network yielded better results in most of the experiments involving different MOT challenges than the other CNN considered in this work, except in the SA case.

Acknowledgment

This research was funded by Tecnologico Nacional de Mexico/ I.T. Chihuahua under grants 5162.19-P and 7598.20-P.

References

[1] H. Yang, S. Qu, C. Chen and B. Yang, "Multiple Objects Tracking With Improved Sparse Representation and Rank Based Dynamic Estimation," IEEE Access, vol. 6, pp. 42264-42278, 2018.

[2] J. Xiang, G. Zhang, and J. Hou, “Online Multi-Object Tracking Based on Feature Representation and Bayesian Filtering Within a Deep Learning Architecture,” IEEE Access, vol. 7, pp. 27923–27935, 2019.

[3] H. Yang, J. Wen, X. Wu, L. He, and S. Mumtaz, “An Efficient Edge Artificial Intelligence MultiPedestrian Tracking Method with Rank Constraint,” IEEE Trans. Ind. Informatics, vol. 15, no. 7, pp. 4178–4188, 2019.

[4] H. Yang, J. Li, J. Liu, Y. Zhang, X. Wu, and Z. Pei, “Multi-Pedestrian Tracking Based on Improved Two Step Data Association,” IEEE Access, vol. 7, pp. 100780–100794, 2019.

[5] S. Hong, T. You, S. Kwak, and B. Han, “Online tracking by learning discriminative saliency map with convolutional neural network,” in 32nd International Conference on Machine Learning (ICML), 2015, pp. 597–606.

[6] J. Zhu, H. Yang, N. Liu, M. Kim, W. Zhang, and M. H. Yang, “Online Multi-Object Tracking with Dual Matching Attention Networks,” in European Conference on Computer Vision (ECCV), 2018, pp. 379–396.

[7] S. H. Bae and K. J. Yoon, “Confidence-Based Data Association and Discriminative Deep Appearance Learning for Robust Online Multi-Object Tracking,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, pp. 595–610, 2018.

[8] H. Kieritz, S. Becker, W. Hubner, and M. Arens, “Online Multi-Person Tracking using Integral Channel Features Hilke,” in 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2016, pp. 122–130.

[9] C. Kim, F. Li, and J. M. Rehg, “Multi-object tracking with neural gating using bilinear LSTM,” in European Conference on Computer Vision (ECCV), 2018, pp. 200–215.

[10] A. Sadeghian, A. Alahi, and S. Savarese, “Tracking the Untrackable: Learning to Track Multiple Cues with Long-Term Dependencies,” in IEEE International Conference on Computer Vision (ICCV), 2017, pp. 300–311.

[11] K. Fang, Y. Xiang, X. Li, and S. Savarese, “Recurrent Autoregressive Networks for Online Multi-object Tracking,” in IEEE Winter Conference on Applications of Computer Vision (WACV), 2018, pp. 466–475.

[12] P. Chu and H. Ling, “FAMNet: Joint Learning of Feature, Affinity and Multi-dimensional Assignment for Online Multiple Object Tracking,” in IEEE International Conference on Computer Vision (ICCV), 2019, pp. 6171–6180.

[13] L. Leal-Taixe, C. Canton-Ferrer, and K. Schindler, “Learning by Tracking: Siamese CNN for Robust Target Association,” in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, pp. 418–425.

[14] H. Sheng, Y. Zhang, J. Chen, Z. Xiong and J. Zhang, "Heterogeneous Association Graph Fusion for Target Association in Multiple Object Tracking," IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 11, pp. 3269-3280, 2019

[15] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, “MOT16: A Benchmark for Multi-Object Tracking,” 2016. [Online]. Available: http://arxiv.org/abs/1603.00831.

[16] C. Ma, J. Bin Huang, X. Yang, and M. H. Yang, “Robust Visual Tracking via Hierarchical Convolutional Features,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 11, pp. 2709–2723, 2019.

[17] W. Li and X. Wang, “Locally aligned feature transforms across views,” in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2013, pp. 3594–3601.