

Ordered Weighted Aggregation Networks for Video Face Recognition

Jacinto Rivero-Hernández^{a,*}, Annette Morales-González^b, Lester Guerra Denis^a,
Heydi Méndez-Vázquez^b

^a Technological University of Havana José Antonio Echeverría, 114th Ave. #11901, Marianao 11500, Havana, Cuba

^b Advanced Technologies Application Center, 7ma A #21406, Playa 12200, Havana, Cuba



ARTICLE INFO

Article history:

Received 15 July 2020

Revised 31 December 2020

Accepted 14 March 2021

Available online 22 March 2021

Keywords:

Neural aggregation network

Ordered weighted average operator

Video face recognition

ABSTRACT

Video face recognition generally includes a step where all descriptors extracted for each frame are aggregated to generate a single video face representation. The most commonly used operator for aggregation is average, which gives the same relevance to each frame. Some adaptive aggregation algorithms have been developed, but most of them rely on the use of weighted mean as aggregation operator, thus disregarding many other types of aggregation operators. In this paper, we propose a novel adaptive aggregation scheme based on ordered weighted average (OWA) operators in contrast with the mainly used weighted mean scheme. Furthermore, besides presenting the theoretical aspects of our aggregation scheme, we develop two different concrete implementations to validate its suitability for video face recognition: Ordered weighted aggregation network (OWANet) and Weighted OWANet (WOWANet). Both algorithms are based on neural networks and are trainable through gradient descent in a classic supervised learning way. We conduct extensive experiments on YouTube Faces, COX Face and the IARPA Janus Benchmark A for evaluating recognition performance on verification and identification tasks. The experimentation process shows that both proposals achieve very competitive results in accuracy with respect to the existent state-of-the-art methods, while significantly reducing space and inference time.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

Video face recognition has gained popularity in security-related scenarios, where several uncontrolled variations of pose, illumination, facial expressions, among others, can drastically reduce the recognition accuracy. Hence, there have been great efforts for finding solutions to deal with these uncontrolled variations, while reducing the computational complexity [1]. In this context, two main solution approaches have emerged to decide which video frames should be used for face recognition: The quality-based approach and the aggregation-based approach.

The quality-based approach is based on the idea that faces with frontal pose, uniform illumination and neutral expression are more representative of the person identity. According to this, quality-based methods select a subset of video frames with lower variations instead of using the whole video for generating face representation. Frame selection can be done through estimation of heuristics like face pose, mouth or eye opening angle and image

resolution among other criteria [2]. Furthermore, each heuristic criterion can be used in individual or combined mode for selecting purposes. However, the quality-based approach presents the following drawbacks: (i) non-selected frames are ignored but they still have relevant information about face identity, (ii) selected frames contribute equally to the final face representation assuming that all selected frames have the same representativeness of the underlying identity and (iii) each heuristic criterion has its own individual estimation error which can induce an accumulated error in the final decision about face quality.

The aggregation-based approach has emerged in [3–7] for overcoming above issues. This is based on learning how to adaptively aggregate information across all video frames for generating a single face representation. Different from the quality-based approach, each frame contributes according to its relevance to the final face representation avoiding equal contributions and preventing accumulated errors of heuristic estimations. In practice, quality-based methods use average operator over an input subset for generating the face representation, while aggregation-based methods use weighted average operator over the whole input for generating the face representation.

The main problem of the aggregation-based approach is that the relevance given by the weighted average operator only takes

* Corresponding author.

E-mail addresses: jacintorivero28@gmail.com (J. Rivero-Hernández), amorales@cenatav.co.cu (A. Morales-González), lguerra@ceis.cujae.edu.cu (L.G. Denis), hmendez@cenatav.co.cu (H. Méndez-Vázquez).

into account individual element relevance ignoring its relative position inside the input set. This leads to negative consequences such as: (i) more parameters are needed for effective aggregation learning, (ii) it cannot incorporate external heuristic information and (iii) the aggregation will have the same behavior always once the model is trained.

There are other operators more flexible than the weighted average, such as the ordered weighted average (OWA). This is still a weighted operator that can perform like maximum, minimum, mean, median and k -order statistics. Furthermore, it uses positional information in the aggregation process and more important, it can combine individual element relevance with positional information. The motivation behind using OWA operator comes from recent investigations on document summarization and key phrase extraction [8,9]. In those papers, OWA and t -norm were successfully applied to aggregation, outperforming other operators. To the best of our knowledge, this is the first proposal based on OWA for aggregation learning in video face recognition. Most existing works are based on weighted average scheme.

In this paper we propose a novel adaptive aggregation scheme that effectively combines OWA operators with aggregation networks serving as a theoretical basis for implementing new algorithms. Furthermore, we develop two concrete implementations of the proposed scheme; the first one is based on pure OWA operator and the second one is based on a combination of pure OWA operator and weighted mean (WOWA).

Summarizing, the main contributions of this paper are:

1. First steps exploring the use of OWA operators for video face recognition.
2. A novel aggregation scheme based on OWA operator that serves as basis for implementing new aggregation algorithms.
3. Two new aggregation algorithms based on the proposed scheme: Ordered weighted aggregation network (OWANet) and Weighted OWANet (WOWANet) that combines OWA and weighted mean.
4. Lightweight aggregation models more efficient than other methods maintaining state-of-the-art accuracy.

Another advantage of our proposal is that it can be used for aggregating other kinds of descriptors, beyond the video face recognition task. For example, on combining different descriptors from distinct biometric traits like in Soft Biometrics. The same idea can be applied to other scenarios such as text mining or signal processing, for combining a set of feature vectors into a single one.

2. Related works

The most important stage in any video face recognition pipeline is to extract the face representation also known as feature vector or descriptor. Since deep learning revolution [10], convolutional neural networks (CNNs) have become the state-of-the-art for extracting face representations, like in many computer vision task [11]. Different methods have been developed on top of deep CNN features for accurate video face recognition; they can be split into classifiers and distance models.

Classifiers treat the problem as supervised classification over large datasets leading to expensive training process. Distance models try to embed the video into a metric space where we can measure the distance or similarity between faces and use it for performing face recognition. These models can be categorized into spatial or aggregation models, depending on how they handle the final face representation for a video sequence.

Spatial models address the video representation as a feature space, i.e., as a whole sparse set of elements located at a specific region in the metric space. In this sense, videos can be represented as probabilistic distributions, affine hulls or manifolds [12,13]. The

main drawback of this approach is that it usually does all-versus-all comparisons for measuring the similarity between features. This is extremely expensive and impractical for applying in real video face recognition scenarios.

On the other hand, the aggregation approach can maintain high accuracy and efficiency simultaneously. It is based on combining all video descriptors into one, which is used as video face representation. The three main stages of video face recognition under this approach are presented as follows:

1. Extracting one descriptor per video frame using a CNN previously trained on still images.
2. Aggregating all extracted descriptors into one compact descriptor using some aggregation operator.
3. Using the aggregated feature vector as final video representation for performing face recognition.

2.1. Aggregation networks

Several aggregation-based algorithms have been proposed in the last years, most of them based on neural networks. For example, Rao *et al.* [14] address aggregation as a Markov decision process trained through reinforcement learning. Discriminative aggregation network (DAN) [15] proposed by Rao *et al.* combines deep metric learning with generative adversarial networks [16]. Quality aware network (QAN) [17] proposed by Liu *et al.* has a two-branch architecture, one for feature extraction and the other one for quality estimation. GhostVLAD [18] proposed by Zhong *et al.* employs a modified NetVLAD [19] layer to down weight the contribution of low quality frames. However, the aforementioned models were designed to be trained in a end-to-end manner, i.e., each model trains its own CNN for feature extraction along with the aggregation learning process. This entangles the aggregation model with a specific CNN and also overloads the training phase increasing time consumption.

Neural aggregation network (NAN) [3] proposed by Yang *et al.* is a deep learning model that aggregates all video descriptors into a single descriptor. The general idea behind this model is to assign weights to each descriptor and then combine them all together through weighted mean. Different from previous models, NAN disentangles the feature extraction and the aggregation stages, working at descriptor level, i.e., it can use any well established CNN previously trained for feature extraction, only relying the training load on the aggregation stage. Furthermore, NAN has served as basis for other aggregation networks and it does not use temporal information, which means that it can be successfully applied to image set recognition as well.

There are different models that extend NAN aggregation scheme. Liu *et al.* [4] proposes a fine-grained neural aggregation network (FG-NAN) that extends NAN aggregation scheme assigning relevance for each descriptor and for each descriptor dimension. FG-NAN similar to NAN does not use temporal information and it works at descriptor level. Gong *et al.* [5] proposes C-FAN as a two-branch network with component-wise feature aggregation. This model also learns a weight vector for each descriptor component but, unlike NAN or FG-NAN, it uses the last convolutional feature map. Gong *et al.* [6] proposes a recurrent embedding aggregation network (REAN) that also incorporates the key features of NAN, but it is aware of temporal information. REAN architecture uses a bidirectional long short-term memory [20] for integrating temporal information in the aggregation step. Gong *et al.* [7] proposes a multi-mode aggregation recurrent network (MARN) that extends REAN model adding multiple linear layers at the end of the model for performing aggregation. As REAN, MARN is aware of temporal information. It should be noticed that REAN and MARN

due to temporal awareness are more suitable for videos than NAN or FG-NAN, but also they are more heavyweight models.

The aggregation scheme of all cited methods is based on weighted mean carrying the aforementioned problems of this operator. Our work is inspired by NAN but we change the aggregation scheme introducing the ordered weighted average operator. Furthermore, our proposal tries to maintain a lightweight aggregation scheme that can be employed for image sets as well.

3. Fundamentals on aggregation operators

From the mathematical point of view, the aggregation problem consists in aggregating n -tuples of objects all belonging to a given set, into a single object of the same set. In our specific case, the objects are feature vectors all belonging to \mathbb{R}^n but for simplicity in this section we explain the main concepts of aggregation operators with the real number set \mathbb{R} . Thus, we define an aggregation operator as a function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ that satisfies: identity when unary, boundary conditions and non decreasing properties.

There are different aggregation operators such as arithmetic mean, maximum, minimum and median but in this paper we mainly cover arithmetic mean, weighted mean and ordered weighted average operators. Other operators are out the scope of this paper. For a comprehensive review on aggregation operators and their properties please refer to [21].

The arithmetic mean (the average) is an aggregation operator that implies equal contributions of each input element to the aggregated value. This operator satisfies the properties of compensatory, monotonicity, continuity, symmetry, associativity, idempotence and stability for linear transformations and it has no behavioral properties.

The weighted mean extends the arithmetic mean allowing to place weights on the arguments. It is mathematically expressed by:

$$f_{w_1, \dots, w_n}(x_1, x_2, \dots, x_n) = \sum_{i=1}^n w_i x_i, \quad (1)$$

where all elements in the weight set $\{w_i\}$ are non negative ($w_i \geq 0$) and their sum is equals one ($\sum_{i=1}^n w_i = 1$). If $\forall w_i \in \{w_i\}, w_i = \frac{1}{n}$ then the weighted mean is reduced to the arithmetic mean. The weighted mean is a function parametrized by $\{w_i\}$ and this peculiarity opens the problem of obtaining the correct weights that remains for all weighted aggregation operators.

The ordered weighted average (OWA) is a family of positional aggregation operators defined by Yager [22] to provide a means for aggregating scores associated with the satisfaction of multiple criteria. It is mathematically expressed by:

$$f_{w_1, \dots, w_n}(x_1, x_2, \dots, x_n) = \sum_{j=1}^n w_j x_{\sigma(j)}, \quad (2)$$

where σ is a permutation that orders the input set $\{x_i\}$ such that: $x_{\sigma(1)} \leq x_{\sigma(2)} \leq \dots \leq x_{\sigma(n)}$ or vice versa. This ordering stage is the key aspect of OWA operators leading to a positional weighted aggregation scheme, i.e., a weight w_j is not being associated with a specific input element x_i but with a particular position j . The weights $\{w_j\}$ remains the properties aforementioned for the weighted mean.

The OWA operator can generalize many of the well-known operators such as the maximum, the minimum, the k -order statistics, the median and the arithmetic mean simply choosing a particular weight set. Yager [23] suggests to use the knowledge of a linguistic quantifier as a way for calculating $\{w_j\}$ and guiding the aggregation process. Specifically, the regular increasing monotone (RIM) quantifiers are the most interesting for [24]. Let Q be a step function that satisfies $Q(0) = 0$, $Q(1) = 1$ and if $x \leq y$ then

$Q(x) \leq Q(y)$; therefore Q is considered a RIM quantifier. On the basis of this kind of quantifiers Yager [22] proposed to compute the weights using the equation:

$$w_j = Q\left(\frac{n-j+1}{n}\right) - Q\left(\frac{n-j}{n}\right) \quad (3)$$

For feature vector aggregation, we need to induce the order of $\{x_i\}$ using another variable that preferably assess the feature vector quality. According to this, the aggregation is made through the induced OWA (IOWA) operator proposed by Yager and Filev [25] as follows:

$$f_{w_1, \dots, w_n}(\langle a_1, x_1 \rangle, \dots, \langle a_n, x_n \rangle) = \sum_{j=1}^n w_j x_{\sigma(j)}, \quad (4)$$

where σ is a permutation that orders the input set $\{x_i\}$ according to the order inducing variable a_i .

4. Proposal

4.1. Proposed aggregation scheme

The overall framework of our aggregation scheme is shown in Figure 1. The proposal takes a face image set of a person as input and outputs a single feature vector as its representation for the recognition task. Let $\mathbf{F} = \{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_N\}$ be the set of CNN feature vectors extracted from the images in a template T , where each \mathbf{f}_k is a D -dimensional vector and N is the number of images in the template. The final face representation \mathbf{r} is aggregated through:

$$\mathbf{r} = \sum_{j=1}^N w_j \mathbf{f}_{\sigma(j)}, \quad (5)$$

where σ is a permutation that arranges \mathbf{F} in descending order according to the order inducing variable $\{a_k\}$ that represents the relevance vector. In this sense, the first element in the ordered set $\{\mathbf{f}_{\sigma(j)}\}$ will be the most relevant and so on. We obtain $\{a_k\}$ employing the attention block defined by Yang et al. [3]. This takes as input a set of feature vectors $\{\mathbf{f}_k\}$ and outputs a relevance value for each feature vector. Thus, if $\{\mathbf{f}_k\}$ is the descriptor set and \mathbf{q} is the kernel of the attention block, then the attention block generates a relevance vector $\{e_k\}$ through dot product as shown in eq. 6. After that, the softmax operator is applied to the relevance vector for generating a normalized relevance vector $\{a_k\}$ with $\sum_k a_k = 1$ according to eq. 7. The process for obtaining the weight set $\{w_j\}$ used in aggregation is discussed in Section 4.2.

$$e_k = \mathbf{q}^T \mathbf{f}_k \quad (6)$$

$$a_k = \frac{\exp(e_k)}{\sum_j \exp(e_j)} \quad (7)$$

Our proposed aggregation scheme satisfies the idempotence, boundary conditions and non decreasing properties though its proof is out the scope of this paper. Nevertheless, we discuss some interesting properties that also satisfy our aggregation scheme. According to eq. 5, it is guaranteed that the aggregated descriptor \mathbf{r} has the same size of a single feature vector \mathbf{f}_k and the number of descriptors in $\{\mathbf{f}_k\}$ does not affect the size of \mathbf{r} . Furthermore, aggregated representation \mathbf{r} is invariant to descriptor order in $\{\mathbf{f}_k\}$ but it is not invariant to descriptor order in the ordered feature set $\{\mathbf{f}_{\sigma(j)}\}$. In addition, the scheme is capable of aggregating input sets with arbitrary number of images. The trainable parameter \mathbf{q} is the attention block kernel and it has the same dimension that a single descriptor \mathbf{f}_k . The attention mechanism is adaptive to the input face generating the relevances in order to the proper input as shown in eq. 6.

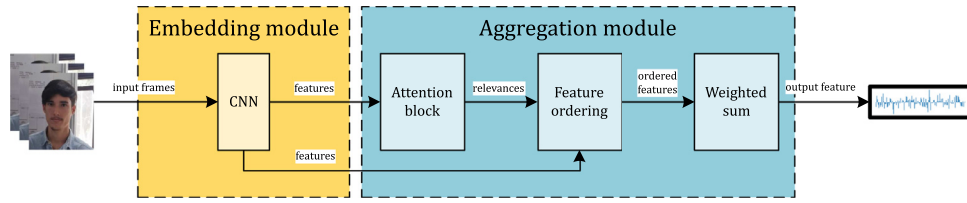


Fig. 1. The proposed aggregation scheme is designed to be a two-module framework. The embedding module extracts the feature vectors from the input frames employing state-of-the-art CNNs, then the aggregation module combines them into a single feature vector. The aggregation is performed in three steps: (1) obtaining a relevance vector through an attention block, (2) ordering the feature set according to the relevance vector and (3) aggregating the ordered feature set using the weights obtained from a linguistic quantifier.


							
Att. block	0.0698	0.0697	0.0692	0.0669	0.0621	0.0659	0.0646
OWANet	0.2582	0.1890	0.1058	0.0691	0.0	0.0	0.0

Fig. 2. Example of differences in weight distributions given by OWANet and a single NAN attention block.

4.2. Ordered weighted aggregation networks

In order to include OWA operator in a learning process, we propose OWANet, which is an aggregation network that uses an individual NAN attention block for estimating feature relevances and it obtains the weight set using a linguistic quantifier according to eq. 3. The elements in the weight set $\{w_j\}$ are all non negative ($w_j \geq 0$) and their sum is equal to one ($\sum_{j=1}^N w_j = 1$). Moreover, the weights are arranged in descending order for ensuring that the first element has the highest relevance value and so on. However, as OWANet obtains the weights through a non-differentiable process, the gradient flow is cropped for training in a classic supervised learning way. In order to address this issue, we implement a dual behavior on OWANet, i.e., we use two distinct aggregation schemes: one differentiable for training and the previously proposed for inference. According to this, OWANet behaves like a single NAN attention block at training time performing the aggregation as follows:

$$\mathbf{r} = \sum_{k=1}^N a_k \mathbf{f}_k \quad (8)$$

At inference time, the aggregation is performed as described in eq. 5.

It is worth noting that OWANet aggregation disregards the real potential of attention block only using it for ordering purposes. Motivated by this, we developed WOWANet as a way of exploiting the whole potential of the attention block along with the OWA-based aggregation. WOWANet extends OWANet but it changes the weight computation process for using the WOWA operator as follows:

$$w_j = Q\left(\sum_{i=1}^j a_{\sigma(i)}\right) - Q\left(\sum_{i=1}^{j-1} a_{\sigma(i)}\right) \quad (9)$$

where σ is a permutation that arranges $\{a_k\}$ in descending order and Q is a RIM quantifier. It should be noticed that WOWANet keeps the dual behavior of OWANet as well as its other properties.

An example of weight distribution assigned by the proposal is shown in Figure 2. As can be seen, the proposal prevents from giving weights to occluded, low resolution and pose variation frames. This means that images with better quality get higher weights. On the other hand a single NAN attention block gives a closer weight distribution.

5. Experiments

5.1. Experimental settings

We conducted experiments over popular databases for evaluating video face recognition algorithms in both verification and identification tasks. YouTube Faces (YTF) [26] is considered a standard database for evaluating video face verification, it groups 3425 videos of 1595 different individuals where each video was recorded from media like TV shows and interviews. COX Face [27] is a database strongly oriented to video surveillance scenario, it contains videos and images of 1000 distinct subjects. Each subject has three different quality videos and one high quality frontal image. The IARPA Janus Benchmark A (IJB-A) [28] is a challenging database for unconstrained face recognition in the wild. It contains 500 subjects with full pose variations, manually localized faces, a wider geographic distribution and a mix of images and videos. For YouTube Faces we evaluated its standard verification protocol, while for COX Face we evaluated its video-to-video, still-to-video and video-to-still identification protocols. In the case of IJB-A, we evaluated its verification protocol.

As a preprocessing step, we employed MTCNN [29] model for detecting and normalizing all face images in all datasets. During the training step, we used five different controllable factors common to all models: frames per video, batch size, training epochs, optimizer and loss function. In the YouTube Faces case, we used 48 frames per video, 384 videos per batch, 20 training epochs, Adam [30] optimizer with learning rate set at 10^{-3} and Contrastive Loss [31] function with margin set at 2.0. In the COX Face and IJB-A cases, we used 21 frames per video, 64 videos per batch, 20 training epochs, Adam [30] optimizer with learning rate set at 10^{-3} and ArcFace [32] loss function. Both OWANet and WOWANet used MostFeng [24] as linguistic quantifier. We employed three distinct state-of-the-art CNN architectures: DLib [33], ShuffleNetV2 [34] and MobileNet [35] for observing the model aggregation performance under different feature extractors. Finally, in all conducted experiment we used the cosine similarity for establishing the vector comparisons. In addition, we have trained and evaluated other state-of-the-art models for a fair comparison: NAN, FG-NAN, REAN and MARN.

5.2. Identification evaluation on COX Face

Table 1 and 2 show the Identification rates at rank-1 in COX Face for still-to-video, video-to-still and video-to-video protocols, respectively. It can be appreciated in Table 1 that in general, our proposals have better identification rates than the other compared models, and in those cases that is outperformed by other models, the results are very closed. It should be noticed that in this case MARN has closed to zero identification rates. This is because MARN aggregation scheme is not idempotent, i.e., it transforms the underlying vector space when it performs the aggregation and this makes that the aggregated video descriptor and the

Table 1
Identification rate at rank-1 (%) reported for COX Face still-to-video and video-to-still protocols.

	DLib					
	S-V1	S-V2	S-V3	V1-S	V2-S	V3-S
Mean	26.60 ± 0.72	40.94 ± 0.98	75.22 ± 0.54	29.30 ± 1.08	45.08 ± 1.23	79.87 ± 0.59
NAN	32.54 ± 6.01	45.75 ± 4.23	78.07 ± 1.22	34.34 ± 4.70	48.17 ± 3.49	80.34 ± 0.76
FG-NAN	27.48 ± 2.18	41.54 ± 1.95	75.40 ± 0.78	30.21 ± 2.04	45.22 ± 2.35	79.68 ± 0.97
REAN	25.35 ± 6.88	37.95 ± 9.76	70.87 ± 10.03	29.04 ± 7.47	42.21 ± 9.58	75.92 ± 8.02
MARN	0.20 ± 0.15	0.17 ± 0.14	0.12 ± 0.12	0.10 ± 0.11	0.11 ± 0.09	0.15 ± 0.12
OWANet	39.94 ± 3.92	53.12 ± 4.03	83.38 ± 1.82	36.57 ± 4.05	48.98 ± 5.26	78.77 ± 1.31
WOWANet	39.35 ± 4.03	52.10 ± 4.29	83.27 ± 2.18	36.51 ± 3.73	48.65 ± 5.07	78.17 ± 1.16
ShuffleNetV2						
	S-V1	S-V2	S-V3	V1-S	V2-S	V3-S
Mean	94.42 ± 0.47	92.64 ± 0.67	99.32 ± 0.16	96.30 ± 0.28	94.70 ± 0.55	99.34 ± 0.18
NAN	96.08 ± 0.49	92.94 ± 0.59	99.34 ± 0.16	97.57 ± 0.44	95.14 ± 0.56	99.41 ± 0.21
FG-NAN	95.14 ± 0.35	92.28 ± 0.62	99.34 ± 0.16	96.98 ± 0.46	95.11 ± 0.58	99.45 ± 0.24
REAN	96.25 ± 0.46	93.08 ± 0.61	99.34 ± 0.20	97.60 ± 0.39	95.50 ± 0.38	99.50 ± 0.22
MARN	0.30 ± 0.19	0.20 ± 0.12	0.24 ± 0.13	0.20 ± 0.12	0.20 ± 0.07	0.22 ± 0.18
OWANet	97.02 ± 0.56	93.78 ± 0.66	99.42 ± 0.19	97.95 ± 0.39	95.41 ± 0.45	99.51 ± 0.23
WOWANet	97.18 ± 0.44	93.87 ± 0.62	99.41 ± 0.15	97.97 ± 0.38	95.40 ± 0.46	99.57 ± 0.26
MobileNetV1						
	S-V1	S-V2	S-V3	V1-S	V2-S	V3-S
Mean	91.11 ± 0.56	92.10 ± 0.77	98.72 ± 0.31	93.87 ± 0.62	92.50 ± 0.41	99.04 ± 0.23
NAN	93.98 ± 0.53	92.47 ± 0.70	98.94 ± 0.30	95.14 ± 0.47	92.94 ± 0.48	99.24 ± 0.21
FG-NAN	92.30 ± 0.54	92.27 ± 0.86	98.87 ± 0.23	94.67 ± 0.58	93.05 ± 0.54	99.11 ± 0.19
REAN	93.27 ± 1.07	92.51 ± 0.84	98.87 ± 0.34	94.97 ± 0.84	93.20 ± 0.61	99.15 ± 0.25
MARN	0.21 ± 0.07	0.15 ± 0.08	0.27 ± 0.17	0.14 ± 0.09	0.27 ± 0.18	0.15 ± 0.12
OWANet	95.35 ± 0.41	93.60 ± 0.49	99.25 ± 0.21	96.71 ± 0.46	93.37 ± 0.52	99.21 ± 0.21
WOWANet	95.30 ± 0.44	93.57 ± 0.57	99.17 ± 0.26	96.70 ± 0.49	93.37 ± 0.28	99.17 ± 0.17

Table 2
Identification rate at rank-1 (%) reported for COX Face video-to-video protocols.

	DLib					
	V1-V2	V1-V3	V2-V3	V2-V1	V3-V1	V3-V2
Mean	41.98 ± 0.64	29.42 ± 0.73	55.38 ± 1.61	45.61 ± 0.95	33.98 ± 0.73	60.17 ± 1.17
NAN	46.58 ± 4.37	35.41 ± 5.71	61.14 ± 6.87	50.22 ± 4.30	40.58 ± 7.40	65.08 ± 5.25
FG-NAN	43.57 ± 2.27	30.90 ± 3.21	56.80 ± 3.86	47.00 ± 1.86	35.24 ± 2.53	61.10 ± 2.98
REAN	43.51 ± 5.98	30.58 ± 6.15	55.21 ± 10.71	48.75 ± 5.34	35.80 ± 5.51	59.31 ± 8.83
MARN	55.05 ± 9.35	46.74 ± 13.89	68.01 ± 11.65	60.04 ± 6.70	51.15 ± 10.40	71.87 ± 6.77
OWANet	49.60 ± 1.87	40.10 ± 4.67	65.35 ± 6.31	51.94 ± 2.91	45.08 ± 3.55	67.21 ± 3.90
WOWANet	49.82 ± 1.19	39.60 ± 4.24	64.68 ± 6.10	52.24 ± 2.08	44.40 ± 3.33	66.94 ± 4.13
ShuffleNetV2						
	V1-V2	V1-V3	V2-V3	V2-V1	V3-V1	V3-V2
Mean	97.68 ± 0.29	99.15 ± 0.15	99.20 ± 0.19	97.77 ± 0.40	98.45 ± 0.32	98.18 ± 0.36
NAN	98.51 ± 0.32	99.54 ± 0.09	99.30 ± 0.20	98.37 ± 0.38	98.82 ± 0.33	98.25 ± 0.32
FG-NAN	98.14 ± 0.32	99.34 ± 0.15	99.22 ± 0.18	97.92 ± 0.36	98.50 ± 0.29	98.17 ± 0.39
REAN	98.64 ± 0.33	99.52 ± 0.09	99.27 ± 0.20	98.62 ± 0.37	98.90 ± 0.34	98.25 ± 0.29
MARN	98.51 ± 0.29	99.12 ± 0.18	99.02 ± 0.25	98.55 ± 0.24	99.00 ± 0.25	98.65 ± 0.27
OWANet	98.71 ± 0.16	99.64 ± 0.12	99.38 ± 0.25	98.70 ± 0.25	99.24 ± 0.22	98.12 ± 0.37
WOWANet	98.65 ± 0.25	99.64 ± 0.07	99.41 ± 0.20	98.64 ± 0.23	99.27 ± 0.23	98.08 ± 0.37
MobileNetV1						
	V1-V2	V1-V3	V2-V3	V2-V1	V3-V1	V3-V2
Mean	96.61 ± 0.61	98.78 ± 0.15	98.92 ± 0.23	97.14 ± 0.36	98.37 ± 0.28	98.48 ± 0.30
NAN	97.72 ± 0.29	99.04 ± 0.17	98.97 ± 0.19	97.97 ± 0.36	98.91 ± 0.24	98.61 ± 0.26
FG-NAN	97.25 ± 0.44	98.90 ± 0.24	98.97 ± 0.19	97.41 ± 0.28	98.61 ± 0.25	98.50 ± 0.31
REAN	97.47 ± 0.73	99.00 ± 0.36	99.00 ± 0.26	97.64 ± 0.66	98.77 ± 0.31	98.48 ± 0.23
MARN	98.07 ± 0.53	99.17 ± 0.18	98.98 ± 0.22	98.14 ± 0.48	98.84 ± 0.34	98.47 ± 0.34
OWANet	98.17 ± 0.40	99.31 ± 0.16	98.94 ± 0.18	98.12 ± 0.35	99.14 ± 0.15	98.87 ± 0.22
WOWANet	98.02 ± 0.39	99.28 ± 0.15	99.02 ± 0.13	98.02 ± 0.22	99.15 ± 0.15	98.88 ± 0.17

image descriptor are not in the same metric space becoming the distance or similarity measure meaningless with the consequent very poor performance. However MARN outperforms other methods in the video-to-video scenario. In particular, it is better than our proposals when the DLib descriptor is used. For ShuffleNetV2 and MobileCosFaceV1 the performances are very similar, but our proposals reached better results than the other models in most cases.

5.3. Verification evaluation on YouTube Faces

Our experimental results on YouTube Faces are shown in Table 3 where we reported the area under the ROC curve (AUC) and the equal error rate (EER) for each CNN feature. As we can appreciate, our proposals seem to have similar performance with respect to state-of-the-art models although there is not statistical evidence to hold the hypothesis of significant differences. Even so,

Table 3
AUC and EER (%) reported for YouTube Faces standard verification protocol.

	DLib		ShuffleNetV2		MobileNetV2	
	AUC	EER	AUC	EER	AUC	EER
Mean	98.23	6.12	97.28	8.48	98.63	4.48
NAN	98.33	5.92	97.33	8.36	97.97	5.84
FG-NAN	98.28	5.96	97.30	8.44	98.65	4.56
REAN	98.29	6.12	96.98	9.08	98.71	4.64
MARN	94.64	12.12	96.20	10.48	97.73	7.04
OWANet	98.34	6.00	97.23	8.52	98.53	4.56
WOWANet	98.35	5.96	97.25	8.56	98.55	4.68

Table 4
Accuracy (%) comparison on YouTube Faces between methods reported in literature and our proposals.

Method	Accuracy (%)
DeepID2 [36]	93.20 ± 0.20
DAN[15]	94.28 ± 0.69
Wen et al. [37]	94.90 ± 0.00
FaceNet [38]	95.52 ± 0.06
QAN[17]	96.17 ± 0.09
C-FAN [5]	96.50 ± 0.90
OWANet	95.86 ± 0.92
WOWANet	96.00 ± 0.88

Table 5
Verification results on IJB-A in terms of TAR@FAR.

	Verification TAR (%)	
	0.1% FAR	1% FAR
Mean	86.43 ± 1.84	92.70 ± 0.96
NAN	83.73 ± 2.95	90.46 ± 1.44
FG-NAN	86.83 ± 1.95	92.76 ± 0.93
REAN	86.46 ± 1.79	92.68 ± 0.98
MARN	86.53 ± 1.34	92.08 ± 0.95
OWANet	84.54 ± 2.29	91.57 ± 1.05
WOWANet	85.07 ± 1.92	91.90 ± 1.03

in no cases our proposals are the worst model in terms of AUC and EER. We selected the features from MobileNetV2 for comparing our models with other results reported in literature. The results, in terms of classification accuracy, can be seen in Table 4. Our methods outperform the majority of related works. C-FAN and QAN exhibit better accuracy than our proposal, but their results are very close.

5.4. Evaluation on IARPA Janus Benchmark A

The experimental results on IJB-A show that it was the most challenging database by far. The verification results are shown in Table 5 where the TAR@FAR was reported. It can be seen that our proposals show very similar behavior than the other aggregation networks. Although in some cases we are not able to achieve the highest results, we believe our methods display an adequate trade-off between accuracy and efficiency, as we will discussed in Section 5.5. A better visual intuition of results on IJB-A can be encountered in Figure 3 where we show the ROC curves for compared models.

We also compared our results with respect to other state-of-the-art methods in Table 6. The proposed models are better than the other models except for C-FAN and QAN. However, those two methods are trained end-to-end entangling the feature extraction step along with the aggregation step, while our proposals can be applied with any feature descriptor.

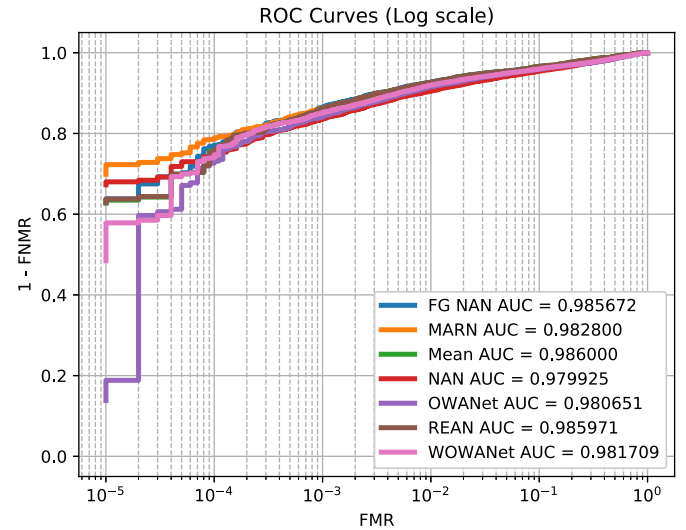


Fig. 3. ROC curves for compared models in IJB-A database.

Table 6
TAR@FAR comparison on IJB-A with state-of-the-art methods.

Method	Verification TAR (%)	
	0.1 % FAR	1% FAR
Shi et al. [39]	60.20 ± 6.90	82.30 ± 2.20
Pooling Faces [40]	63.10 ± 0.00	81.90 ± 0.00
Pose-aware Models [41]	65.20 ± 3.70	82.60 ± 1.80
Masi et al. [42]	72.50 ± 0.00	88.60 ± 0.00
Triplet Embedding [43]	81.30 ± 2.00	90.00 ± 1.00
QAN [17]	89.30 ± 3.90	94.20 ± 1.50
C-FAN [5]	91.59 ± 0.99	93.97 ± 0.78
OWANet	84.54 ± 2.29	91.57 ± 1.05
WOWANet	85.07 ± 1.92	91.90 ± 1.03

Table 7
Average aggregation time for CPU and GPU (in milliseconds), amount of parameters and storage size (in bytes) for the different methods.

	Aggregation time		Model size	
	CPU	GPU	Parameters	Storage
REAN	26282.00	1864.20	9 060 480	34 500 000
MARN	4925.50	362.31	215 168	844 000
FG-NAN	59.70	21.87	33 024	129 000
NAN	9.37	23.44	16 640	65 600
OWANet	4.68	1.56	128	968
WOWANet	4.68	1.56	128	968

5.5. Efficiency analysis

In order to analyze the efficiency of our proposal, we took into account two factors: aggregation time and number of parameters in the models. We defined our own protocol for gathering the aggregation times. Firstly, we generated a random set of 10K feature vectors being each one a 128-dimensional vector. Secondly, each model aggregates the random set 10 times in each processing unit (CPU or GPU). All experiments were conducted on a desktop computer Intel Core i5-7500 equipped with a graphics processing unit NVIDIA 1050 Ti. The average results are reported in Table 7 along with the model size. It can be seen that our proposals are more lightweight than the other state-of-the-art models, specially for those LSTM-based. As a consequence, OWANet and WOWANet have the smallest aggregation times in both CPU and GPU by far. Lastly, it may be noticed that NAN model has larger aggregation time on GPU than on CPU. This is an interesting and counterintuitive phenomenon but still can happen in some cases and it can be

due to the fact that the training is made on CPU as it is mentioned in [3].

A final comment: it can be seen in Table 3, 1, 2, 4 and 5 that our proposals OWANet and WOWANet share very similar behavior in performance. WOWANet is a natural extension of OWANet, and even if it does not have an apparent improvement over OWANet, we consider that the theoretical basis of it (the combination of ordering and weighted mean) can be useful for future related researches.

6. Conclusions

In this work we explored the use of a novel aggregation scheme based on OWA operators for video face recognition. We proposed OWANet and WOWANet as two new lightweight adaptive aggregation algorithms supported over the theoretical basis provided by our OWA-based aggregation scheme. The experimental results showed that the proposals have better accuracy in still-to-video and video-to-still protocols for identification task. In other settings the proposals achieve very similar recognition accuracy compared with other state-of-the-art models. However, the improvements of our proposals are in efficiency where they reached smaller aggregation times than the other compared models while maintaining state-of-the-art accuracy. The lightness of our proposals make them a suitable option to consider for applying in real video face recognition scenarios. Our future works are oriented to develop a full differentiable OWA-based aggregation scheme.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] V.A. Kumar, S. Ramya, H. Divakar, G.K. Rajeswari, A survey on face recognition in video surveillance, in: International Conference on ISMAC in Computational Vision and Bio-Engineering, Springer, 2018, pp. 699–708.
- [2] H. Méndez-Vázquez, L. Chang, D. Rizo-Rodríguez, A. Morales-González, Evaluación de la calidad de las imágenes de rostros utilizadas para la identificación de las personas, *Computación y Sistemas* 16 (2) (2012) 147–165.
- [3] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, G. Hua, Neural aggregation network for video face recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4362–4371.
- [4] Z. Liu, H. Hu, J. Bai, S. Li, S. Lian, Fine-grained attention-based video face recognition, *CoRR* (2019).
- [5] S. Gong, Y. Shi, N.D. Kalka, A.K. Jain, Video face recognition: Component-wise feature aggregation network (c-fan), in: 2019 International Conference on Biometrics (ICB), IEEE, 2019, pp. 1–8.
- [6] S. Gong, Y. Shi, A.K. Jain, N.D. Kalka, Recurrent embedding aggregation network for video face recognition, 2019 arXiv preprint arXiv:1904.12019.
- [7] S. Gong, Y. Shi, A. Jain, Low quality video face recognition: Multi-mode aggregation recurrent network (MARN), in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2019.
- [8] E. Valladares-Valdés, A. Simón-Cuevas, J.A. Olivas, F.P. Romero, A fuzzy approach for sentences relevance assessment in multi-document summarization, in: International Workshop on Soft Computing Models in Industrial and Environmental Applications, Springer, 2019, pp. 57–67.
- [9] M. Barreiro-Guerrero, A. Simón-Cuevas, Y. Pérez-Guadarrama, F.P. Romero, J.A. Olivas, Applying OWA Operator in the Semantic Processing for Automatic Keyphrase Extraction, in: Iberoamerican Congress on Pattern Recognition, Springer, 2019, pp. 62–71.
- [10] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, *Nature* 521 (7553) (2015) 436–444.
- [11] G. Guo, N. Zhang, A survey on deep learning based face recognition, *Computer Vision and Image Understanding* 189 (2019) 102805.
- [12] Y. Hu, A.S. Mian, R. Owens, Sparse approximated nearest points for image set classification, in: CVPR 2011, IEEE, 2011, pp. 121–128.
- [13] K.-C. Lee, J. Ho, M.-H. Yang, D. Kriegman, Video-based face recognition using probabilistic appearance manifolds, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1, Citeseer, 2003, pp. 1–313.
- [14] Y. Rao, J. Lu, J. Zhou, Attention-aware deep reinforcement learning for video face recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3931–3940.
- [15] Y. Rao, J. Lin, J. Lu, J. Zhou, Learning discriminative aggregation network for video-based face recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3781–3790.
- [16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, *Advances in neural information processing systems* 14 (2014) 2672–2680.
- [17] Y. Liu, J. Yan, W. Ouyang, Quality aware network for set to set recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5790–5799.
- [18] Y. Zhong, R. Arandjelović, A. Zisserman, GhostVLAD for set-based face recognition, in: Asian Conference on Computer Vision, Springer, 2018, pp. 35–50.
- [19] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, J. Sivic, NetVLAD: CNN architecture for weakly supervised place recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 5297–5307.
- [20] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, *Neural networks* 18 (5–6) (2005) 602–610.
- [21] M. Detyniecki, B. Bouchon-meunier, R. Yager, H. Prade, Mathematical aggregation operators and their application to video querying, UC Berkeley, 2000 Ph.D. thesis.
- [22] R.R. Yager, On ordered weighted averaging aggregation operators in multicriteria decisionmaking, *IEEE Transactions on systems, Man, and Cybernetics* 18 (1) (1988) 183–190.
- [23] R.R. Yager, Quantifier guided aggregation using owa operators, *International Journal of Intelligent Systems* 11 (1) (1996) 49–73.
- [24] L. Feng, T.S. Dillon, Using fuzzy linguistic representations to provide explanatory semantics for data warehouses, *IEEE Transactions on Knowledge and Data Engineering* 15 (1) (2003) 86–102.
- [25] R.R. Yager, D.P. Filev, Induced ordered weighted averaging operators, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 29 (2) (1999) 141–150.
- [26] L. Wolf, T. Hassner, I. Maoz, Face recognition in unconstrained videos with matched background similarity, in: CVPR 2011, IEEE, 2011, pp. 529–534.
- [27] Z. Huang, S. Shan, R. Wang, H. Zhang, S. Lao, A. Kuerban, X. Chen, A benchmark and comparative study of video-based face recognition on cox face database, *IEEE Transactions on Image Processing* 24 (12) (2015) 5967–5981.
- [28] B.F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, A.K. Jain, Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1931–1939, doi:10.1109/CVPR.2015.7298803.
- [29] J. Xiang, G. Zhu, Joint face detection and facial expression recognition with MTCNN, in: 2017 4th International Conference on Information Science and Control Engineering (ICISCE), IEEE, 2017, pp. 424–427.
- [30] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014 arXiv preprint arXiv:1412.6980.
- [31] R. Hadsell, S. Chopra, Y. LeCun, Dimensionality reduction by learning an invariant mapping, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006, pp. 1735–1742.
- [32] J. Deng, J. Guo, N. Xue, S. Zafeiriou, Arcface: Additive angular margin loss for deep face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 4690–4699.
- [33] D.E. King, Dlib-ml: A machine learning toolkit, *Journal of Machine Learning Research* 10 (Jul) (2009) 1755–1758.
- [34] N. Ma, X. Zhang, H.-T. Zheng, J. Sun, Shufflenet v2: Practical guidelines for efficient cnn architecture design, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 116–131.
- [35] S. Chen, Y. Liu, X. Gao, Z. Han, Mobilefacenet: Efficient cnns for accurate real-time face verification on mobile devices, in: Chinese Conference on Biometric Recognition, Springer, 2018, pp. 428–438.
- [36] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in: Advances in neural information processing systems, 2014, pp. 1988–1996.
- [37] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: European conference on computer vision, Springer, 2016, pp. 499–515.
- [38] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: A unified embedding for face recognition and clustering, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815–823.
- [39] Y. Shi, A. Jain, Improving face recognition by exploring local features with visual attention, in: 2018 International Conference on Biometrics (ICB), IEEE, 2018, pp. 247–254.
- [40] T. Hassner, I. Masi, J. Kim, J. Choi, S. Harel, P. Natarajan, G. Medioni, Pooling faces: Template based face recognition with pooled face images, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2016, pp. 59–67.
- [41] I. Masi, S. Rawls, G. Medioni, P. Natarajan, Pose-aware face recognition in the wild, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 4838–4846.
- [42] I. Masi, A.T. Trn, T. Hassner, J.T. Leksut, G. Medioni, Do we really need to collect millions of faces for effective face recognition? in: European conference on computer vision, Springer, 2016, pp. 579–596.
- [43] S. Sankaranarayanan, A. Alavi, C.D. Castillo, R. Chellappa, Triplet probabilistic embedding for face verification and clustering, in: 2016 IEEE 8th international conference on biometrics theory, applications and systems (BTAS), IEEE, 2016, pp. 1–8.