Adversarial Deep Multi-Task Learning Using Semantically Orthogonal Spaces and Application to Facial Attributes Prediction

Arnaud Dapogny, Gauthier Tallec, Jules Bonnard, Edouard Yvinec, Kevin Bailly

Adversarial Deep Multi-Task Learning Using Semantically Orthogonal Spaces and Application to Facial Attributes Prediction

Arnaud Dapogny[1], Gauthier Tallec[2], Jules Bonnard[2], Edouard Yvinec[1,2] and Kévin Bailly[1,2]
[1] Datakalab, 114 Boulevard Malesherbes, 75017 Paris, France
[2] Sorbonne Université, 4 Place Jussieu, 75005 Paris, France

***Abstract*— Deep learning-based multi-task approaches usually rely on factorizing representation layers up to a certain point, where the network splits into several heads, each one addressing a specific task. Depending on the inter-task correlation, such naive model may or may not allow the tasks to benefit from each others. In this paper, we propose a novel Semantic Orthogonality Spaces (SOS) method for multi-task problems, where each task is predicted using the information from a common subspace that factorizes information among all tasks, as well as a task-specific subspace. We enforce orthogonality between these tasks by applying soft orthogonality constraints, as well as adversarially-learned semantic orthogonality objectives that ensures that predicting one task requires the specific information related to that task. We demonstrate the effectiveness of SOS on synthetic data, as well as for large-scale facial attributes prediction. In particular, we use SOS to craft a lightweight architecture that provides high-end accuracies on CelebA database.**

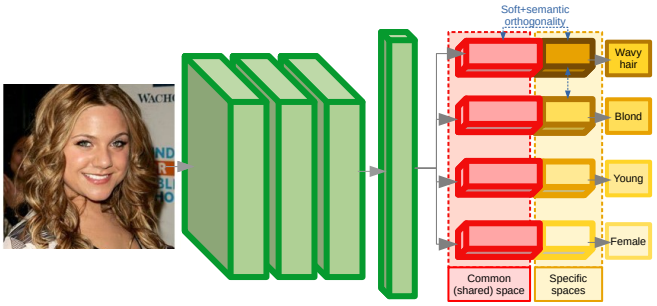

Fig. 1. Overview of the proposed SOS multi-task method and application to facial attributess prediction. A *common* and several task-*specific* spaces are derived from the representations layer to predict each attribute. Soft and semantic orthogonality constraints are applied to efficiently factor and disentangle information.

I. INTRODUCTION

Deep Multi-task learning refers to the process of predicting multiple non-exclusive values, corresponding to as many tasks, using a single network. It is an ubiquitous paradigm in machine learning and computer vision, as it allows to compensate for a lack of training data, to a certain extent. As such, it finds a wide number of applications that range from image classification [10] or semantic segmentation [2] to facial attributes [13] or action unit detection [4]. The usual way to integrate several tasks inside a deep neural network consists in sharing representations between tasks up to a certain point. From this common representation space, multiple heads corresponding to the various tasks are appended. However, as reported in [10], there is no guarantee that learning shared representations benefits either task. Even worse, there is no way to reliably know beforehand whether such a multi-task model will perform better than single-task ones. We argue that this problem might stem from the fact that such representation does not promote factorization of a common information as well as specificity of the different tasks.

To address this problem, we propose a new multi-task formulation, which is based on Semantically Orthogonal Spaces (SOS), as illustrated on Figure 1. In SOS, each task is predicted by combining a *common* and task-*specific* space. To sum it up, the contributions of this paper are three-folds:

This work has been supported by Datakalab as well as the French National Agency (ANR) in the frame of its Technological Research JCJC program (FacIL, project ANR-17-CE33-0002).

- We introduce a novel Semantically Orthogonal Spaces (SOS) multi-tasks formulation, which is based on combining *common* and task-*specific* spaces, and consists in incorporating soft and semantic orthogonality constraints involving adversarial objectives.
- We show that SOS performs better that traditional multi-task approaches on toy experiments on several datasets, as well as for large-scale facial attributes prediction.
- We integrate SOS multi-task prediction into a squeeze-and-excitation-like deep architecture, which achieves state-of-the-art results for facial attributes prediction with a lightweight architecture.

II. RELATED WORK

In this section, we review existing deep multi-task learning approaches, as well as their applications in a facial attributes prediction context.

a) Multi-task learning: Surveys on deep multi-task learning methods can be found in [23], [19]. Generally speaking [4], [3], multi-task learning aims at compensating for the lack of training data, by sharing the weights of a deep network between different task, so that these tasks can benefit from each other. To do that, a naïve approach [10] consists in simply using a different prediction head for each task, sharing the representation layers up to a certain point. However, such approach intrinsically presents a number of problems. First, as pointed out in [16] there is no way to know beforehand which representation layers should or should not be shared across tasks. To adress this problem, authors in [16] propose to use cross-stitch units that merge

together the representation power of multiple architectures. Second, this parallel prediction order may not be optimal [15], and learning more complicated prediction dependencies may improve the overall performance. For instance, Meyerson *et al.* [15] design a soft ordering pipeline, in which the network can dynamically select the best sequence of tasks to be predicted, through multiple stages. Tallec et al. [20] propose to learn an optimal task chaining order along with the task prediction itself, in a joint manner. Zamir *et al.* [22] propose to define a taxonomic structure based on inter-task correlation. This structure represents an order in which visual tasks shall be combined to enhance the prediction accuracy with limited amounts of data. Finally, the final shared representation needs to factor inter-task common information while preserving task-specific information in order to find an underlying structure between the tasks at stake. However, a naïve multi-task approach [10] may not promote this: as a result, loosely related tasks can "contaminate" each other, which is bound to cause performance drops. In the work of Nicolle *et al.* [17] in the frame of facial action unit detection, the authors use a concatenation of a *common* and *specific* space to predict each task. However, this method does not explicitely promote orthogonalization between the spaces, hence the possibility that *specific* spaces contaminate each other still exists. By contrast, we propose to integrate soft orthogonality constraints to limit the coupling of these spaces, as well as a novel semantic orthogonality term involving adversarial objectives.

b) Attribute prediction: Facial attributes prediction is an interesting case of multi-task problem. As introduced in [13], it consists in classifying 40 attributes from registered face images. These attributes are heterogeneous and some of them are highly correlated (e.g. *male* and *beard* or wearing hearrings or *blond hair* and *bald*) while some others hardly are (e.g. *young* and *smiling*). Furthemore, facial attributes datasets are generally highly imbalanced [6], thus successful multi-task methods shall capture these inter-tasks correlations despite high dataset biases. In [13], the authors propose to use a combination of two deep networks that first precisely localize the face region of interest, then performs attribute prediction from it. By contrast, Gunther *et al.* [5] propose an alignment-free procedure, along with a dedicated data augmentation scheme to enhance prediction. Walk & learn [21] explore the use of contextual information for attribute prediction. MOON [18] proposes a mixed objective optimization network with a domain adaptive loss weighting to address the imbalance problem. Hand *et al.* [7] add an auxiliary network on top of a multi-task deep network attribute prediction to better model the inter-task relationships. Later on, the same authors [6] also proposed a batch balancing method for improving attribute prediction, allowing them to achieve high-end accuracies with few parameters. Kalayeh *et al.* [9] propose to use semantic segmentation of the face image to enhance attribute prediction. Lu *et al.* [14] propose an adaptive task grouping process that allows to generate a dependency tree among related attribute prediction task. This leads to networks structures where a particular attribute can benefit from other ones predicted downstream in the network, in a similar vein as [15], [22].

III. Methodology

Figure 2 illustrates the flowchart of our method, in the case of a multi-task problem with only 2 tasks. A traditional, naïve multi-task approach (Section III-A) is showcased on the left, while our soft orthogonal spaces method (Section III-B) is depicted on the right. This method uses soft orthogonality constraints (Section III-C) as well as semantic orthogonality constraints (Section III-D) to learn task-specific embeddings as well as common inter-task representation. Finally, Section III-E introduces the attribute prediction problem, which is a classical multi-task learning application.

A. A naïve multi-task formulation

In what follows, we consider a dataset $(x_j, y_i^*)_{j=1...m, i=1...n}$ with m examples and n non mutually-exclusive, possibly correlated tasks. Generally speaking, a traditional, naïve multi-task approach consists in using a shared embedding $h_j = g_l \circ ... \circ g_1(x_j)$ to predict the n tasks from an example x_j. $g_1, ..., g_l$ denote a number of transformations of the input x_j, e.g. a number of CNN or fully-connected layers, max pooling, batch normalization, ReLU activation function. Estimates for the n tasks are thus computed from the shared embedding h_j by applying:

$$\hat{y}_{ji} = t_i(h_j) \tag{1}$$

Where the t_i are parametric functions such as CNN or fully-connected layers. Such basic multi-task model is usually learned by optimizing the following loss:

$$L_{sup}(\Theta) = \sum_{j=1}^{m} \sum_{i=1}^{n} \mathcal{L}(\hat{y}_{ji}, y_{ji}^*) \tag{2}$$

With Θ the set of parameters of the network, and $\mathcal{L}$ being any usual loss function, e.g. sigmoid cross-entropy, $\mathcal{L}_1$ or $\mathcal{L}_2$ loss, depending on the application. Such basic multi-task approach usually offers advantages over a single-task approach, where each task is predicted using an independant network and set of representations, as the networks may learn better representations, allowing each task to benefit from the others while limiting overfitting. This, however, is not guaranteed at all, and, in certain cases, the performance of a multi-task model can be lower than that of a single-task one [10].

B. Common and task-specific spaces

To address this problem, we propose to split the representation h into $n+1$ representation spaces, the first of which being a *common* representation h^c, that models the inter-task relationships. The subsequent ones are task-*specific* spaces h_i^s that contain information that can not be factorized into the *common* representation. Formally, for an example j the representation can be written as $h = h^c||h_{j1}^s||...||h_{jn}^s$, with $||$ the concatenation operator. A task i can then be predicted using a concatenation of the *common* and *specific* vectors relatively to this task.

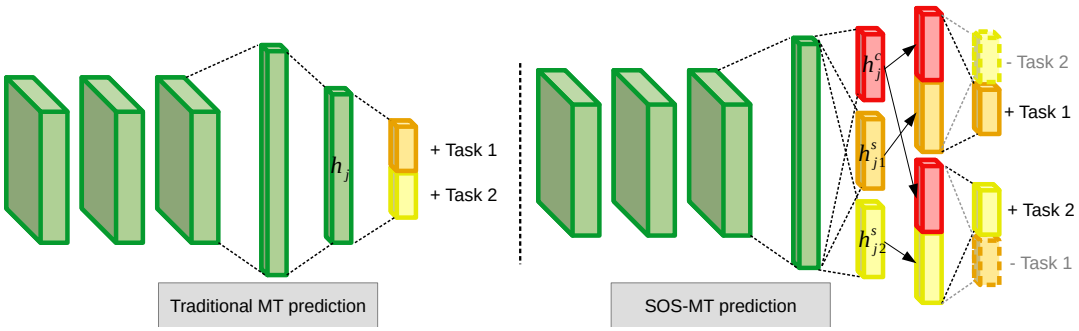

Fig. 2. Illustration of the proposed method in case of a two-tasks problem. Left: traditional MT prediction is usually performed by sharing parameters between two tasks until the end of the network, where each task correspond to one dimension of an output vector. Right: by contrast, the proposed SOS-MT uses a common space that is shared between the two tasks and two specific spaces, one for each task. Orthogonality between common and specific spaces and the two specific spaces is promoted by soft ortogonality losses between these embeddings (blue/violet arrows), as well as inter-task adversarially-learned semantic orthogonality loss.

$$\hat{y}_{ji} = t_i(h^c||h^s_{ji}) \tag{3}$$

A special case of this formulation is when $dim(h^s_{ji}) = 0 \; \forall i$: this corresponds to the naïve formulation seen in Section III-A. Compared to this naïve formulation, using a combination of a common subspace and multiple task-specific subspaces allows in theory to factor the inter-task information and to predict each task using specialized features, hence capturing task complementarity. At this point, however, nothing prevents the task-specific subspaces to be correlated (**problem 1**). Even worse, there is a possibility that the *common* subspace encode all the information while leaving the *specific* subspaces nearly useless (**problem 2**).

C. Soft orthogonality constraints

We address **problem 1** by using soft orthogonality constraints to limit the correlation between, one the one hand, the *common* and *specific* subspaces and, one the other hand, the different *specific* subspaces. Towards this purpose, as it was done [1] for domain adaptation, we add a regularization coefficient on the Frobenius norm of the inner product of these representations. This can be written:

$$L^{soft}_{c \leftrightarrow s}(\Theta) = \frac{1}{n} \sum_{j=1}^{m} \sum_{i=1}^{n} ||h^c.h^s_{ji}||^2_F \tag{4}$$

for limiting *common-specific* coupling, and:

$$L^{soft}_{s \leftrightarrow s}(\Theta) = \frac{1}{n(n-1)} \sum_{j=1}^{m} \sum_{i=1}^{n} \sum_{k=1, k \neq i}^{n} ||h^s_{ji}.h^s_{jk}||^2_F \tag{5}$$

To limit *specific-specific* coupling between the tasks. In the special case where the h^s_{ji} are centered with unit variance (which can be forced e.g. *via* using batch normalization), this amounts to explicitly penalizing correlation between the variables. However, we still need to make sure that the *common* subspace does not encore all the information, by forcing poor predictions for a task when not using the corresponding *specific* subspace.

D. Semantic orthogonality via adversarial learning

To adress **problem 2** we need to ensure that without the *specific* subspace h^s_{ji}, the model fails to predict task i. This can be done in two ways: first, we can push every subspace $h^c||h^s_{jk}$, $k \neq i$ away from correctly predicting task i. To do that, we minimize the following loss:

$$L^{sem}(\Theta) = \frac{-1}{n(n-1)} \sum_{j=1}^{m} \sum_{i=1}^{n} \sum_{k=1, k \neq i}^{n} \mathcal{L}(\tilde{y}^k_{ji}, y^*_{ji}) \tag{6}$$

Where $y^k_{ji} = t^k_i(h^c||h^s_{jk})$ is a "fake" prediction of task i label using only information from the *common* subspace and the task-k-*specific* subspace. For instance t^k_i can be a very simple prediction head, e.g. a single fully-connected layer with parameters $\tilde{\Theta}$. While this formulation may allow to ensure that the i-th *specific* subspace is needed to predict task i, in this formulation nothings prevents t^k_i to degenerate, in which case no gradient is backpropagated anymore. To avoid this pitfall, we add an adversarial objective on the optimization of these "fake" predictions:

$$L^{adv}(\tilde{\Theta}) = \frac{1}{n(n-1)} \sum_{j=1}^{m} \sum_{i=1}^{n} \sum_{k=1, k \neq i}^{n} \mathcal{L}(\tilde{y}^k_{ji}, y^*_{ji}) \tag{7}$$

In this formulation, the $n(n-1)$ fake prediction heads can be seen as as many discriminators that try to predict the correct attributes without using the related *specific* subspaces, while the multi-task network is optimized to ensure that such prediction is not possible. The total loss is:

$$\begin{aligned} L(\Theta, \tilde{\Theta}) = L_{sup}(\Theta) + \lambda_{c \leftrightarrow s} L^{soft}_{c \leftrightarrow s}(\Theta) + \lambda_{s \leftrightarrow s} L^{soft}_{s \leftrightarrow s}(\Theta) \\ + \lambda_{adv} L^{sem}(\Theta) + \lambda^{disc}_{adv} L^{adv}(\tilde{\Theta}) \end{aligned} \tag{8}$$

In what follows, we define our Semantic Orthogonal Spaces (SOS) module as the multi-task prediction head introduced in Section III-B, optimized with the loss function defined in Equation (8). More specifically, we denote SOS(C,S) an SOS module with a C-dimensional common subspace, and

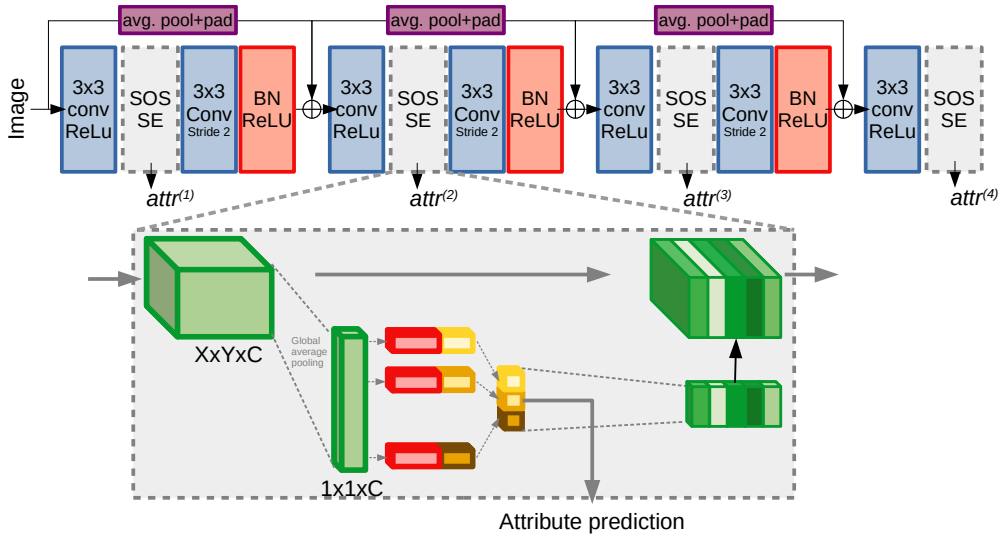

Fig. 3. SOS-SE architecture. It is composed of several stacked blocks (4 on the illustration), each containing *conv*, SOS MT SE, *conv* and BN layers. The SOS MT SE layer is a squeeze-and-excitation (SE) layer which uses attribute prediction with SOS-MT as its bottleneck (squeeze) layer. Stacking SOS MT SE blocks allows the tasks to benefit from each other, in the vein of [15].

S-dimensional task-specific subspaces. This brick is generic and replace the multi-task prediction head to address any multi-task problem, as it will be seen in Section IV.

E. Application to attribute prediction

Facial attributes prediction is a natural choice of application for multi-task methods, as it consists in estimating a number of attributes from a face image, some of which are strongly correlated while some are not. A naive way to apply SOS for facial attributes prediction would be to replace the final layer of any backbone CNN by a SOS layer with roughly the same amount of parameters. While this works to a certain extent, as it will be shown in the experiments, we can design a lighter, and overall more efficient architecture that we call SOS-SE, which is illustrated on Figure 3.

SOS-SE network is composed of four blocks, each one of the form conv $\rightarrow$ SOS-MT-SE $\rightarrow$ conv $\rightarrow$ Batch norm/ReLU. The SOS-MT-SE block is composed of a global average pooling (squeeze) layer, a SOS-MT attribute prediction block (with a 512-dimensional *common* space and 40 32-dimensional task-*specific* spaces), and an excitation layer that takes as input the 40 dimensional attribute vector to generate C excitation values with a sigmoid activation, C being the number of input channels. We also add a residual connection between the input and output of each SOS-MT-SE block. The idea of such a block is to (a) use recent advances in squeeze-and-excitation [8] networks, *i.e.* use global statistics on the image to select relevant channels, and (b) to use attribute prediction to drive the learning of such high-level squeeze layers and (c) define a potent architecture with few parameters (less than $3M$) that can be learned from scratch for attribute prediction.

It should also be observed that, in its first blocks, SOS-SE learns to predict attributes from high level image statistics (by constraining the squeeze layer with intermediate supervision of attribute prediction), and to leverage these predictions to select relevant feature maps for predicting attributes using downstream feature maps (using the excitation layers). By doing so, it intrinsically learns a suitable order in which the attributes can help predict each other, similarly to [15]. However, in our approach, the deep representation learning and task ordering schemes are closely entwined by the SE channel selection procedure.

IV. Experiments

First, in Section IV-A we demonstrate the effectiveness of our method in the frame of toy experiments using small deep networks. Then, in Section IV-B we apply it in a large-scale, real-world facial attribute prediction context.

A. Toy experiments

In Section IV-A.1 we provide implementation details to ensure reproducibility. Then in Section IV-A.2 we apply our method on synthetic data. Finally, in Section IV-A.3 we benchmark our method on MNIST and CIFAR databases.

1) Experimental setup: We compare three different architectures for addressing these 2 tasks: **ST** a single-task architecture composed of two independant networks, with $2 \rightarrow 20 \rightarrow 10 \rightarrow 1$ units, with ReLU activations unless for the last layer, which has sigmoïd activation. **MT** a single multi-task architecture with $2 \rightarrow 20 \rightarrow 20 \rightarrow 2$ units and the same activations. and **SOS** whose architecture is $2 \rightarrow 20 \rightarrow SOS(15, 5)$. All the models are trained with the same hyperparameters: we apply 50000 updates of ADAM optimizer with learning rate $5e^{-4}$ (with polynomial annealing), $\beta_1 = 0.9$ and batch size 128. For SOS model we apply $\lambda_{c \leftrightarrow s} = \lambda_{s \leftrightarrow s} = \lambda_{adv} = \lambda_{adv}^{disc} = 0.01$. For the experiments on MNIST, CIFAR-10 and CIFAR-100 we use a LeNet-5 backbone with 3 convolutional layers and 2 FC layers for each model. we train the models with batch size 100 and 200 epochs using ADAM and a constant learning rate

of $2e^{-4}$. For **SOS** model we use $\lambda_{c\leftrightarrow s} = 0.01$ $\lambda_{s\leftrightarrow s} = 0.01$ and $\lambda_{adv} = \lambda_{adv}^{disc} = 0.05$.

2) Experiments on synthetic data: we generate a number of random datasets $\{(X_j, Y_i)\}_{j=1...m, i=1,...,n}$ with $X_j \in [-1,1]^2$ $\forall j$ and $Y_{ji} \in \{0,1\}^2$ $\forall j, i$. We set $m = 5000$ (number of examples) and $n = 2$ (number of tasks). The sets sets are composed of 500 examples drawn from the same distribution as the train sets. To generate the data, we assign each point a class according to the region of the $[-1,1]^2$ interval where this point belongs, and we alternate sampling between the positive and negative classes for each task for both the train and test sets. Based on this process we generate datasets based on combinations of the following two task templates:

- $donut(r, x_0, x_1, \epsilon)$: $r - \epsilon/2 < \sqrt{(X_j^0 - x_0)^2 + (X_j^1 - x_1)^2} < r + \epsilon/2$
- $cos(a, b, c, \epsilon)$: $|X_j^1 - c.cos(a.X_j^0 + b)| < \epsilon/2$

Thus, depending on the parameters r, x_0, x_1, ϵ for the first template, and a, b, c, ϵ for the second one, we can generate two-tasks toy datasets with varying overlap between the tasks. Figure 4 shows the 10 generated datasets that we use in our experiments.

Table I showcases the results obtained for the three models on the 10 toy datasets illustrated on Figure 4. Generally speaking, the basic multi-task model MT performs better than the two single-task models ST on the datasets with strong overlap between the tasks (1,2,6,10), while ST performs better elsewhere (3,4,5,6,9). Note however that this is not always the case, e.g. on dataset 7 where the tasks do not present any overlap, and MT still performs better in this scenario. Overall, in terms of overall accuracy on the 10 datasets, ST performs better than MT, indicating that sharing the weights between the different tasks may not be beneficial in all cases, as also echoed in [10]. The proposed SOS multi-task model performs significantly better than MT in all tested scenarios, and significantly better than ST on all datasets except 4. The overall accuracy accross the 10 datasets is 95.0 for SOS vs 90.5 for ST and 89.7 for MT. Thus, overall, our SOS multi-task method appears as a more efficient way to share weights and disentangle the different tasks as compared to the naïve MT approach, as it allows to factor representations between the two tasks within its *common* subspace, and to complement this representation *via* its task-*specific* subsets. As such, it enables overlapping tasks to benefit from each other, as well as to limit the inter-task interference in case of loosely overlapping tasks.

3) Experiments on MNIST and CIFAR databases: We study the behavior of multi-task (**MT** and **SOS**) models, and performance compared to a single-task (**ST**) model in case of (a) unrelated tasks, and (b) highly correlated tasks. To do so, we regard handwritten character recognition from MNIST images as one task, and object recognition on CIFAR datasets as the other task. When training on MNIST and CIFAR-10 (a), we observe a peformance drop of 1.56% for **MT** model on CIFAR-10 and 0.81% on MNIST, and and 0.31% and 0.68% for **SOS**, respectively. *A contrario*, when training on CIFAR-10 and CIFAR-100 (b), parameter sharing improves the performance by 1.55% on CIFAR-10 and 1.42% on CIFAR-100 for **MT**, and 1.41% on CIFAR-10 and 1.80% on CIFAR-100 for **SOS**. This indicates that thanks to its soft and semantic orthogonalisation properties, the proposed SOS formulation allows to (a) help mitigate the performance drop in case of unrelated task, and (b) allows the tasks to better compliment each other in case of highly related tasks. In the general case, e.g. on large scale experiments involving a variety of tasks, it is generally not possible to know beforehand which task benefits from each other or not, thus SOS appears as an overall more efficient multi-task formulation, as it will be shown in what follows.

B. Large-scale experiments: application to facial attributes prediction

We validate our approach on a large-scale, real world scenario of facial attributes prediction. The **CelebA database [13]** is a large-scale facial attributes database which contains 202599 218×178 celebrity images coming from 10177 identities, each annotated with 40 binary attributes (such as *gender*, *eyeglasses*, *smile*), and the localization of 5 landmarks (nose, left and right pupils, mouth corners). In our experiments, we use the train partition that contains 162770 images from $8k$ identities to train our models. The test and val partitions each contains 19962 instances from roughly $1k$ identities that are different from the training set identities. In Section IV-B.1 we perform ablation study to show the interest of the different components of SOS. Finally, in Section IV-B.2 we compare SOS-SE with state-of-the-art approaches and show that it better captures inter-task correlations.

1) Ablation study: In this first experiment, we perform ablation study by comparing SOS with a baseline MT formulation. We also consider various architectural and hyperparameter settings. To do that, we only consider 128×128 grayscale images extracted from the train partition of CelebA and measure the average unweighted and weighted accuracies (trace of the 2×2 confusion matrix averaged over all attributes) for each model. Results are reported in Table II.

a) Backbone architecture: we append our SOS MT prediction head at the end of a very simple 8-layers backbone CNN with strided convolutions, Batch normalization, ReLU activation and $32 \rightarrow 64 \rightarrow 64 \rightarrow 128 \rightarrow 128 \rightarrow 256 \rightarrow 256 \rightarrow 512$ channels. The outputs are then flattened, fed into a 1024-dimensional fully-connected layer and passed into a SOS layer, with various hyperparameters.

b) Loss hyperparameters: first, we compare SOS with a *common* space of size 512 and 40 *specific* subspaces (one for each attribute) to a baseline network with a single fully-connected layer with the same total number of parameters (corresponding to SOS with a *common* space of size 1792 and no *specific* subspaces). As such, simply using different subspaces for each attribute steadily improves the accuracy, as it already allows to decorrelate predictions and factor relevant information into the *common* subspace. Moreover, using soft orthogonality constraints between the subspaces also improve the accuracy, as it encourages the inter-task

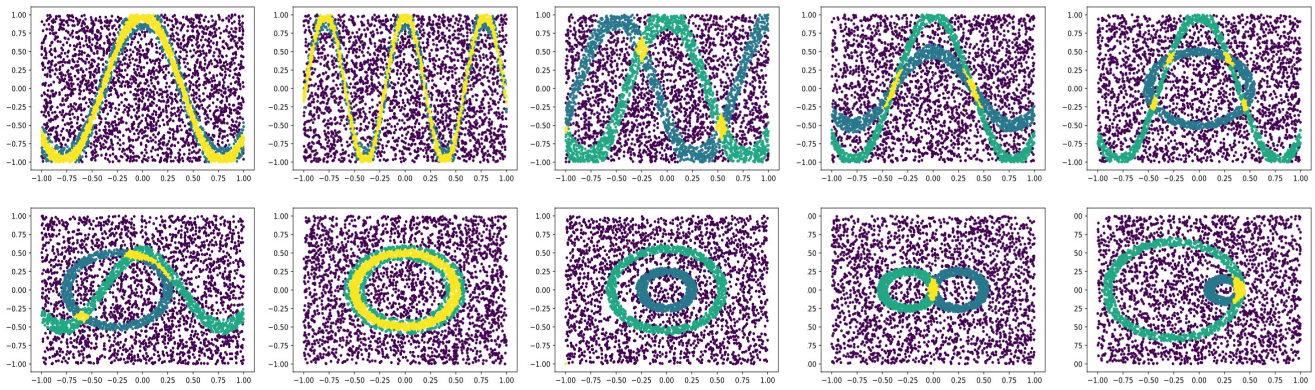

Fig. 4. Randomly generated toy datasets. For each of the 10 datasets, points in violet indicate negative examples for the two tasks (−task1,−task2). Green points: +task1,−task2. Blue points: −task1,+task2. Yellow: +task1,+task2. For certain datasets (e.g. 1,2 and 7) there is a strong overlap between the two tasks, as indicated by the large yellow area.

TABLE I

AVERAGE ACCURACY ON 10 SYNTHETIC TOY TWO-TASKS DATASETS WITH THREE ARCHITECTURES: SINGLE-TASK (ST), NAÏVE MULTI-TASK (MT) AND OUR SEMANTIC ORTHOGONAL SPACES (SOS) MULTI-TASK MODEL. FOR EACH TASK, THE BEST ACCURACY IS REPORTED IN RED, AND THE SECOND BEST IN BLUE.

data	Task 1	Task 2	Accuracy (task 1, %)			Accuracy (task 2, %)			Accuracy (avg, %)		
			ST	MT	SOS	ST	MT	SOS	ST	MT	SOS
1	Cos(4,0,1,0.2)	Cos(4,0,1,0.1)	98.6	98.8	99.2	96.0	96.0	97.6	97.3	97.4	98.4
2	Cos(8,0,1,0.1)	Cos(8,0,1,0.2)	71.1	76.6	80.6	83.6	83.6	91.2	77.4	80.1	85.9
3	Cos(4,2,1,0.2)	Cos(4,0,1,0.2)	90.0	89.2	93.8	98.0	78.4	98.6	94.0	83.8	96.2
4	Cos(4,0,0.5,0.1)	Cos(4,0,1,0.1)	96.8	95.4	97.4	89.2	75.0	78.8	93.0	85.2	88.1
5	Donut(0.5,0,0.1)	Cos(4,0,1,0.1)	97.4	97.4	98.8	79.2	79.2	86.6	88.3	88.3	92.7
6	Donut(0.5,-0.25,0.1)	Cos(4,0,0.5,0.1)	90.4	84.2	97.0	92.0	88.6	92.2	91.2	86.4	94.6
7	Donut(0.5,0,0.1)	Donut(0.5,0,0.2)	94.8	97.4	98.8	95.6	94.6	95.0	95.2	96.0	96.9
8	Donut(0.25,0,0.1)	Donut(0.55,0,0.1)	98.8	98.8	98.8	72.4	98.8	99.4	85.6	98.8	99.1
9	Donut(0.25,-0.25,0.1)	Donut(0.25,0.25,0.1)	96.0	97.4	99.4	93.6	70.6	97.4	94.8	84	98.4
10	Donut(0.15,-0.25,0.1)	Donut(0.65,0.25,0.1)	99.6	99.4	99.6	77.8	95.4	99.2	88.7	97.4	99.4
Average(all tasks)									90.5	89.7	95.0

TABLE II

ABLATION STUDY ON CELEBA TEST SET FOR VARIOUS COMBINATIONS OF ARCHITECTURAL AND HYPERPARAMETER SETTINGS.

architecture	common	specific	$\lambda_{c\leftrightarrow s}$	$\lambda_{s\leftrightarrow s}$	λ_{adv}	λ_{adv}^{disc}	accuracy (%)
CNN8→FC2	1792	0	0	0	0	0	88.30
CNN8→SOS	512	32	0	0	0	0	88.85
CNN8→SOS	512	32	0.01	0	0	0	89.03
CNN8→SOS	512	32	0.01	0.01	0	0	89.10
CNN8→SOS	512	32	0.01	0.01	0.05	0	89.56
CNN8→SOS	512	32	0.01	0.01	0.05	0.05	**89.97**
CNN8→SOS	32	44	0.01	0.01	0.05	0.05	89.12
CNN8→SOS	256	40	0.01	0.01	0.05	0.05	89.04
CNN8→SOS	512	32	0.01	0.01	0.05	0.05	**89.97**
CNN8→SOS	1024	20	0.01	0.01	0.05	0.05	89.56
CNN8→SOS	1472	8	0.01	0.01	0.05	0.05	89.51

decorrelation (by setting $\lambda_{s\leftrightarrow s} > 0$) as well as common information factorization ($\lambda_{c\leftrightarrow s} > 0$). Lastly, using semantic orthogonalization with an adversarial objective provides a significant boost in accuracy, as it ensures that each *specific* subspace is needed to correctly predict each attribute. However, as it is often the case with adversarial learning, setting high values for λ_{adv}^{disc} leads to instability: in our experiments, values from 0.01 to 0.05 provided the best results.

c) Balancing common and specific subspaces: as pointed out above, setting dimension to 0 for *specific* (simple fully-connected layer case) leads to bad results as compared to a SOS layer. The inverse is also true: if the *common* space is too small (size 32 or 256), the model cannot factor inter-task relevant information, which leads to lower accuracies. In our experiments, we witnessed better results with a *common* space of size 512 and *specific* subspaces of size 32, which seems to be a good trade-of between task factorization and task-specific information encoding.

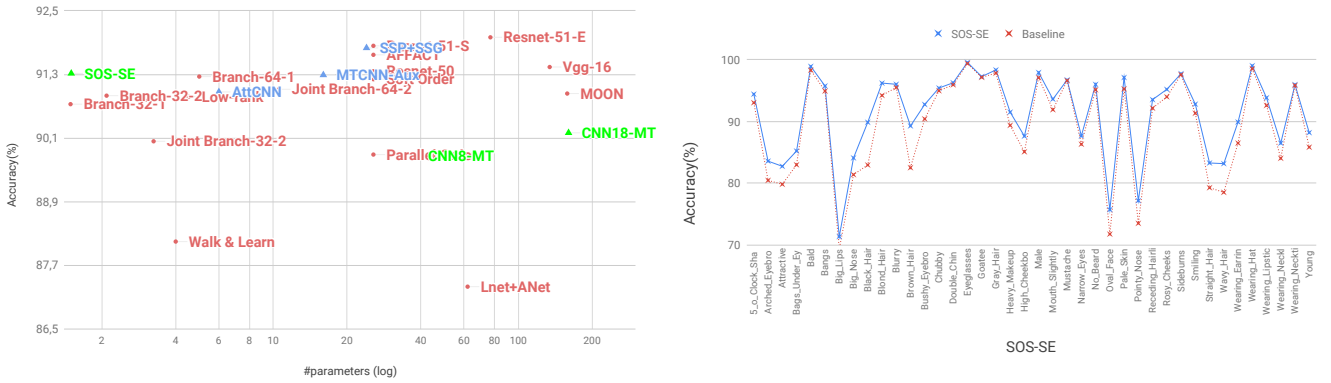

Fig. 5. **Left:** accuracy (%) vs number of parameters plot on CelebA test set, and comparison with state-of-the-art approaches, as reported in the respective papers. Red: models reported in the literature based on pre-trained networks. Blue: Models reported in the literature trained from scratch on CelebA. Green: baseline and SOS-SE models pre-trained from scratch on CelebA. **Right:** per-attribute accuracy (%) of our SOS-SE model (blue) and comparison with a baseline approach (red).

2) Comparison with state-of-the-art techniques: Figure 5 (left) shows a comparison of the results obtained using our SOS-SE model, against results reported in recent papers for facial attributes prediction [13], [21], [18], [14], [5], [7], [15], [9], [6]. It should be empathized that SOS-SE achieves a high-end accuracy of **91.32%**, while using only $\approx 1.5M$ parameters. Compare for instance with state-of-the-art methods AttCNN [6] (90.97%, $\approx 6M$ parameters), MOON [18] (90.94%, $\approx 158M$ parameters) or soft order [15] (91.36%, ≈ 25.6 parameters). In addition, since SOS-SE architecture is very light, it can be trained from scratch on CelebA, contrarily to many approaches that either use AlexNet/VGG/ResNet backbones pre-trained on ImageNet [13], [5], [15] or pre-train on related face verification databases [21], [18]. Moreover, SOS-SE is competitive with recent approaches (e.g. FairGRAPE [11] achieving 90.90%) involving heavy transformer architectures (such as [12] 91.93%). It is also better than baselines CNN8-MT and CNN18-MT (whose architecture mimics VGG19) models trained from scratch on CelebA. Thus, our SOS multi-task formulation using soft and semantic orthogonality constraints via adversarial learning, along with recent architectural design such as squeeze-and-excitation [8], allows to find a good tradeof between model complexity and performance in a large scale facial attributes prediction scenario. Figure 5 (right) shows the per-attribute accuracy obtained with SOS-SE, as well as a comparison with a baseline CNN-8 model trained from scratch on CelebA, in a naïve multi-task fashion. It should be noted that SOS-SE is better than this baseline model for every attribute, while having a lot less parameters.

Figure 6 shows correlations between attributes computed for the ground truth labels and the prediction of both a naive (CNN8-MT) approach and SOS-SE model. A high Pearson correlation Coefficient (PCC) does not necessarily indicate that that two vectors vary in a similar way, e.g. in the case where data is extremely imbalanced, as in CelebA test set. A way to address this issue is to weight each positive examples by $1/s$, where s is the ratio of negative examples for that

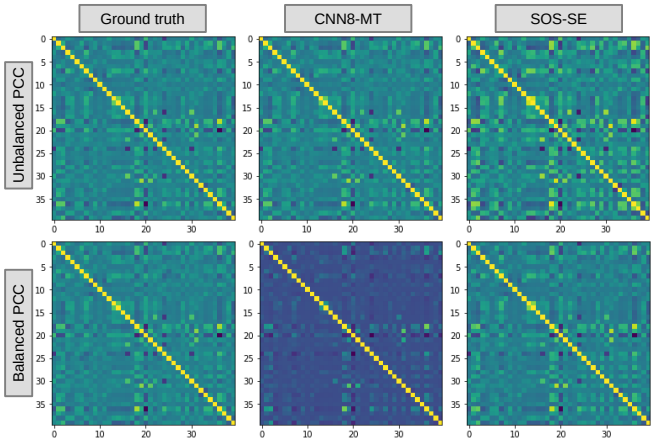

Fig. 6. Inter-task (attribute) correlation matrices for ground truth, naive multi-task and SOS model.

attribute in the test set. This, in turn, allows (minority) positive examples to have more weight in the resulting balanced PCC score. Naturally, the ground truth correlation matrix is the same in both configurations. For the naive approach, the inter-attributes correlation is very close to the ground truth configuration in the unbalanced PCC metric. However, this is not true when we use the balanced PCC score, and vice-versa for SOS-SE. This indicates that the naive model is very sensitive to dataset bias, whereas the SOS model intrinsically captures more correct correct attribute correlations, thanks to the use of task-specific spaces, as well as soft and semantic orthogonality constraints.

V. Discussion and Conclusion

In this paper, we introduced a novel semantic orthogonal spaces (SOS) way to tackle multi-task problems that consists in predicting each task with a combination of a *common* and *specific* space dedicated to this task. Furthermore, we enforce specificity of the task-*specific* spaces by using a combination of soft orthogonality constraints, as well as a novel semantic orthogonality term, that involves adversarial

training. Throughout extensive experiments involving several architectures on synthetic data, small-scale datasets as well as large scale scenarios, we showed that SOS enhances the predictive capacities of deep multi-task learning models, as compared to traditional approaches. In particular, SOS allows to mitigate the performance drop caused by sharing parameters between loosely correlated tasks, as well as to better capture the inter-task correlations between related ones. Last but not least, we showed the interest of SOS for facial attributes recognition in the wild, and used it to design a state-of-the-art, lightweight SOS-SE architecture. Future work involve using SOS inside more complicated network architectures, e.g. those similar to [16], to investigate the effect of applying semantic orthogonalization to structures deeper than a single neuron layers. Furthermore, we plan to apply it to other computer vision domains, ranging from facial action unit prediction to body pose estimation or semantic segmentation.

REFERENCES

[1] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In NeurIPS, 2016.

[2] Y. Chen, A. Dapogny, and M. Cord. Semeda: Enhancing segmentation precision with semantic edge aware loss. arXiv preprint arXiv:1905.01892, 2019.

[3] A. Dapogny, K. Bailly, and S. Dubuisson. Confidence-weighted local expression predictions for occlusion handling in expression recognition and action unit detection. International Journal of Computer Vision.

[4] A. Dapogny, K. Bailly, and S. Dubuisson. Multi-output random forests for facial action unit detection. In FG), 2017.

[5] M. Günther, A. Rozsa, and T. E. Boult. Affact: Alignment-free facial attribute classification technique. In International Joint Conference on Biometrics, 2017.

[6] E. M. Hand, C. Castillo, and R. Chellappa. Doing the best we can with what we have: Multi-label balancing with selective learning for attribute prediction. In AAAI, 2018.

[7] E. M. Hand and R. Chellappa. Attributes for improved attributes: A multi-task network utilizing implicit and explicit relationships for facial attribute classification. In AAAI, 2017.

[8] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In CVPR, 2018.

[9] M. M. Kalayeh, B. Gong, and M. Shah. Improving facial attribute prediction using semantic segmentation. In CVPR, 2017.

[10] I. Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In CVPR, 2017.

[11] X. Lin, S. Kim, and J. Joo. Fairgrape: Fairness-aware gradient pruning method for face attribute classification. arXiv, 2022.

[12] D. Liu, W. He, C. Peng, N. Wang, J. Li, and X. Gao. Transfa: Transformer-based representation for face attribute evaluation. arXiv, 2022.

[13] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In ICCV, 2015.

[14] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, and R. Feris. Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification. In CVPR, 2017.

[15] E. Meyerson and R. Miikkulainen. Beyond shared hierarchies: Deep multitask learning through soft layer ordering. ICLR, 2018.

[16] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. Cross-stitch networks for multi-task learning. In CVPR, 2016.

[17] J. Nicolle, K. Bailly, and M. Chetouani. Facial action unit intensity prediction via hard multi-task metric learning for kernel regression. In FG workshops, 2015.

[18] E. M. Rudd, M. Günther, and T. E. Boult. Moon: A mixed objective optimization network for the recognition of facial attributes. In ECCV, 2016.

[19] S. Ruder. An overview of multi-task learning in deep neural networks. arXiv preprint arXiv:1706.05098, 2017.

[20] G. Tallec, A. Dapogny, and K. Bailly. Multi-order networks for action unit detection. IEEE Transactions on Affective Computing, 2022.

[21] J. Wang, Y. Cheng, and R. Schmidt Feris. Walk and learn: Facial attribute representation learning from egocentric video and contextual data. In CVPR, 2016.

[22] A. R. Zamir, A. Sax, W. Shen, L. J. Guibas, J. Malik, and S. Savarese. Taskonomy: Disentangling task transfer learning. In CVPR, 2018.

[23] Y. Zhang and Q. Yang. A survey on multi-task learning. arXiv preprint arXiv:1707.08114, 2017.