

Tensor-based Subspace Factorization for StyleGAN

René Haas, Stella Graßhof and Sami S. Brandt
IT University of Copenhagen, Copenhagen, Denmark

Abstract—In this paper, we propose τ GAN a tensor-based method for modeling the latent space of generative models. The objective is to identify semantic directions in latent space. To this end, we propose to fit a multilinear tensor model on a structured facial expression database, which is initially embedded into latent space. We validate our approach on StyleGAN trained on FFHQ using BU-3DFE as a structured facial expression database. We show how the parameters of the multilinear tensor model can be approximated by Alternating Least Squares. Further, we introduce a stacked style-separated tensor model, defined as an ensemble of style-specific models to integrate our approach with the extended latent space of StyleGAN. We show that taking the individual styles of the extended latent space into account leads to higher model flexibility and lower reconstruction error. Finally, we do several experiments comparing our approach to former work on both GANs and multilinear models. Concretely, we analyze the expression subspace and find that the expression trajectories meet at an apathetic face that is consistent with earlier work. We also show that by changing the pose of a person, the generated image from our approach is closer to the ground truth than results from two competing approaches.

I. INTRODUCTION

In this paper, we propose a novel framework for finding semantic directions in the latent space of Generative Adversarial Networks (GANs) [10]. GANs have, since their proposal, emerged as one of the most dominant approaches for unsupervised representation learning in Computer Vision and beyond [23].

Architecturally GANs refer to the simultaneous training of two neural networks: a *generator* and a *discriminator*. The generator produces images by sampling from its latent space, while the discriminator, a binary classifier, tries to discriminate the generated images from the training images. The goal of training is to reach the equilibrium of the min-max game between the two adversaries, such that neither can improve by changing the parameter values. At equilibrium, the discriminator can be discarded, and the generator can then be used to produce new data by sampling from the latent distribution. The new data points follow the same statistics as the training data but are not contained in it. Modern state-of-the-art GAN variations have borrowed from the Style-transfer literature [14], [22] to disentangle the latent space and synthesize high-quality face images. Work by [17], [18], and most recently [16], showed how to train state-of-the-art StyleGAN model, even in cases of limited data.

A recent goal has been to find semantically interpretable directions in GAN latent spaces, and several approaches for *semantic face editing* have been proposed. Semantic face editing refers to the ability to change various semantic

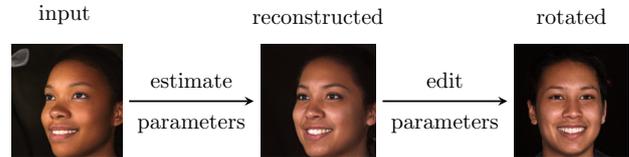


Fig. 1: Overview of the proposed approach.

attributes, such as identity, expression, and rotation, gender, of the generated images. Early work used an information criterion (InfoGAN) [6] to determine semantic directions. However, as pointed out in [8], there is no guarantee that the latent codes produced by this method are semantically meaningful. Additional unsupervised approaches for finding semantic directions in StyleGAN include Principal Component Analysis (PCA) on sampled latent codes [15] and the closed-form factorization suggested by [25].

A recent approach for finding semantic directions in StyleGAN in a supervised fashion is to train binary linear classifiers (SVMs) to detect single binary semantic attributes such as smile vs. no smile, male vs. female, glasses vs. no glasses. For a given semantic attribute, the semantic direction could then be defined as the normal to the supporting hyperplanes of the trained SVM [24].

In the literature, a wide collection of *multilinear* methods have been proposed to model and analyze faces and expressions. Early, PCA or dictionary-based 3D Morphable Models (3DMM) [3], [9] capture the variation in shape and texture of neutral 3D faces. Recently 3DMMs have also been used to make semantic edits to images generated by StyleGAN [27]. More recently, factorization methods, based on higher-order data representations, were introduced with the benefit of better disentanglement of dimensions, such as person-specific shape and expression, when compared to matrix methods [28], [30]. These models were built on the Higher-Order Singular Value Decomposition (HOSVD) to factorize the data, and have successfully been used to model faces, their 3D reconstruction, as well as in transferring expressions [4], [5]. Moreover, in [11], [12] a HOSVD tensor model was constructed from the Binghamton 3D facial expression database (BU-3DFE) [33], which revealed a practically planar expression subspace, in which the six basic emotions form one-dimensional affine subspaces [11]. These six lines intersect in a common vertex, the origin of expressions, which surprisingly does not represent the neutral face, but an extrapolated expression referred to as *apathetic*.

The main novelty of this work is to use a multilinear face model to analyze the latent space of GANs. More specifically, we propose to use the HOSVD to factorize the

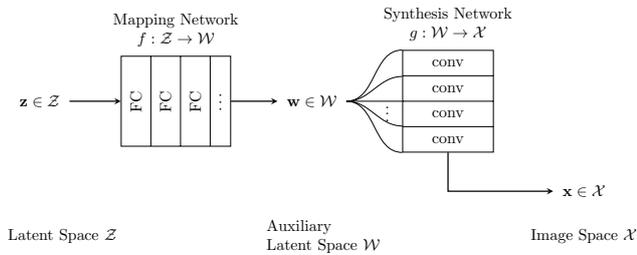


Fig. 2: Architecture of the StyleGAN generator.

latent space into semantically meaningful linear subspaces that yield a multilinear tensor model. Given an input image, we estimate the model parameters to approximate the input, and then change one attribute, such as rotation, as illustrated in Fig. 1.

The main contributions of this paper are as follows:

- We propose a novel method for semantic face editing with StyleGAN.
- We propose a method to estimate model parameters and present reasonable regularization, enabling stable parameter transfer.
- We show that expression trajectories intersect at a unique point, corresponding to the origin of expressions, which differs from the neutral face confirming the earlier findings [11], [12] based on BU-3DFE.
- We propose an extended model, based on style separation, which leads to greater model flexibility and lower reconstruction error for independent test images.

The paper is organized as follows: In Sec. II we will review the architecture and outline the process on how to embed reference images into the latent space of StyleGAN. In Sec. III we present our Tensor-Based GAN model which we build "on top of" the StyleGAN latent space. Here we will also elaborate on how we can approximate model parameters for a given latent vector. Experiments and results of our proposed approach are presented in Sec. IV, followed by a summary and conclusion in Sec. V.

II. STYLEGAN

In this section, we will review the StyleGAN architecture and explain how to embed reference images into the latent space of the pre-trained models released by Nvidia [17], [18].

A. StyleGAN Architecture

The StyleGAN generator $G: \mathcal{Z} \rightarrow \mathcal{X}$, where $G = g \circ f$, is composed of two networks, the mapping network $f: \mathcal{Z} \rightarrow \mathcal{W}$ and the synthesis network $g: \mathcal{W} \rightarrow \mathcal{X}$, see Fig. 2. The mapping network f , maps the latent vector $\mathbf{z} \in \mathcal{Z}$ onto the auxiliary latent space \mathcal{W} to the vector $\mathbf{w} = f(\mathbf{z})$ while the synthesis network $g: \mathcal{W} \rightarrow \mathcal{X}$ maps the vector $\mathbf{w} \in \mathcal{W}$ to the final output image $\mathbf{x} \in \mathcal{X}$ in image space. The full generator G thus maps the latent vector \mathbf{z} to an image \mathbf{x} . The notation used in this paper is summarized in Tab. I.

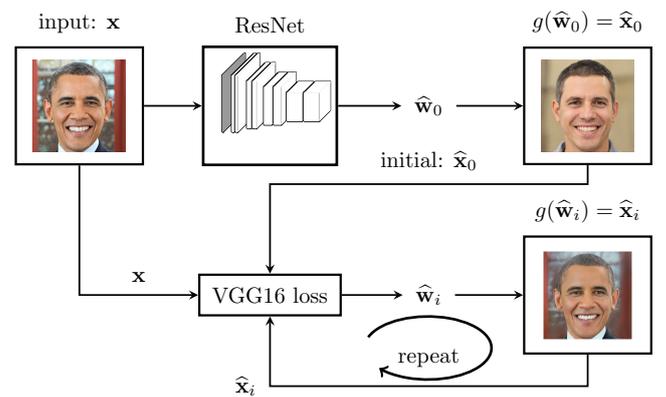


Fig. 3: Diagram illustrating image embedding into the auxiliary latent space \mathcal{W} .

B. Generator Inversion

GANs do not include an encoder as part of their architecture. Therefore, a goal in GAN research has been to find a method for finding a latent code that produces an image as close as possible to a given reference image, which we refer to as *embedding* an image into the latent space. The problem can be considered as inverting the synthesis network $g^{-1}: \mathcal{X} \rightarrow \mathcal{W}$ [1], [21] while inverting G , and thereby embedding into \mathcal{Z} space, has been investigated in [18]. Contemporary techniques for \mathcal{W} space embedding, i.e. finding g^{-1} , use a VGG network [26]. Our approach for embedding onto the auxiliary latent space \mathcal{W} is illustrated in Fig. 3. The inverse generator $G^{-1}: \mathcal{X} \rightarrow \mathcal{Z}$ yields the latent vector $\mathbf{z} = G^{-1}(\mathbf{x})$ with $G^{-1} = f^{-1} \circ g^{-1}$ for the input image \mathbf{x} .

The initial estimate for the auxiliary latent vector for a given reference image is computed as follows. We use the pre-trained weights of StyleGAN [17] and the recently revised StyleGAN2 [18] architecture. Then, as proposed in [2], we train a ResNet [13] in a supervised setting using synthetic StyleGAN data to approximate g^{-1} that yields the initial estimate $\hat{\mathbf{w}}_0$ for the latent vector. The refinement for the auxiliary latent vector is computed by first using the VGG16 network [26], pre-trained on ImageNet database, and then removing the classification layer, hence the truncated network produces a high dimensional feature vector for a given input image, as described in [34]. Since the trained generator is fully differentiable, the loss can be calculated in VGG space and gradients back-propagated through the generator, hence we can iteratively update the latent code. This approach is also used in [21]. We also found that using the ResNet estimate as initialization for the VGG optimization process, leads to faster convergence than not using ResNet initialization.

III. MULTILINEAR MODEL

This section introduces τ GAN, our latent space factorization method for GANs that augments the StyleGAN synthesis network g with a multilinear tensor model. We do this by embedding a facial expression database into the

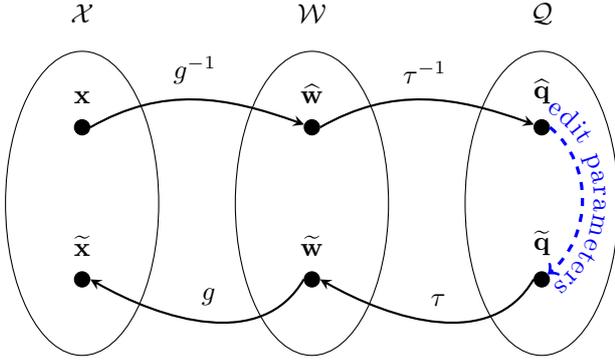


Fig. 4: Overview of the different spaces and how the function relate them, c.f. Tab. I. The blue line indicates a manual change of one of the parameter vectors for transfer of person, expression or rotation.

TABLE I: Overview of the notation used in this work.

Symbol	Description
\mathcal{X}	Image Space
\mathcal{Z}	Latent Space
\mathcal{W}	Auxiliary Latent Space
\mathcal{Q}	Parameter Space
Operator	Name
$f : \mathcal{Z} \rightarrow \mathcal{W}$	Mapping Network
$g : \mathcal{W} \rightarrow \mathcal{X}$	Synthesis Network
$g^{-1} : \mathcal{X} \rightarrow \mathcal{W}$	StyleGAN Embedder
$\tau^{-1} : \mathcal{W} \rightarrow \mathcal{Q}$	Parameter Estimator
$\tau : \mathcal{Q} \rightarrow \mathcal{W}$	Tensor Model

auxiliary latent space \mathcal{W} of StyleGAN. We then order the embedded database into a tensor, which we factorize into semantic subspaces. The resulting parameter space \mathcal{Q} will thus be the Cartesian product of the semantic subspaces $\mathcal{Q} = \mathcal{Q}_P \times \mathcal{Q}_E \times \mathcal{Q}_R$, where \mathcal{Q}_P is the person space, \mathcal{Q}_E the expression space, and \mathcal{Q}_R is the rotation subspace. An overview of the different spaces and how the operators relate them are displayed in Fig. 4 and Tab I.

A. Tensor Factorization

The Higher-Order Singular Value Decomposition (HOSVD) is a generalization of the matrix SVD to higher-order tensors [7], [32], [11], [29], [28], [19].

The starting point for our analysis is a standardized data tensor $T \in \mathbb{R}^{N \times P \times E \times R}$, where N refers to the number of elements in the latent vector, P is the number of people, E the number of expressions, and R number of viewpoints or rotations. Using the HOSVD T can then be factorized as

$$T \simeq C \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3 \times_4 \mathbf{U}_4, \quad (1)$$

where \times_k denotes the k -way product, $C \in \mathbb{R}^{\tilde{N} \times \tilde{P} \times \tilde{E} \times \tilde{R}}$ is the core tensor, and $\mathbf{U}_1 \in \mathbb{R}^{N \times \tilde{N}}$, $\mathbf{U}_2 \in \mathbb{R}^{P \times \tilde{P}}$, $\mathbf{U}_3 \in \mathbb{R}^{E \times \tilde{E}}$, $\mathbf{U}_4 \in \mathbb{R}^{R \times \tilde{R}}$ are matrices with orthonormal columns constructed from the singular vectors of the k -mode matrix unfoldings of T . In general we have that $\tilde{N} \leq N$, $\tilde{P} \leq P$, $\tilde{E} \leq E$, and $\tilde{R} \leq R$.

B. Multilinear Tensor Model for GANs

The HOSVD (1) factorizes the data tensor into a core tensor, and a set of factor matrices \mathbf{U}_i , one for each subspace. By selecting appropriate rows from \mathbf{U}_i , $i = 2, 3, 4$, one normalized latent vector, i.e. a single mode-1 fiber of T , can be recovered. For example, to recover the latent vector of person p performing expression e with rotation r , the p^{th} row of \mathbf{U}_2 , e^{th} row of \mathbf{U}_3 , and r^{th} row of \mathbf{U}_4 is selected. This can be conveniently formulated by a canonical basis, where the parameter vectors $\mathbf{q}'_2 \in \mathbb{R}^P$, $\mathbf{q}'_3 \in \mathbb{R}^E$ and $\mathbf{q}'_4 \in \mathbb{R}^R$ pick a weighted linear combination of the rows of the \mathbf{U}_i matrices. Therefore, a given latent code \mathbf{y}' can be approximated by the model prediction $\hat{\mathbf{y}}'$ as

$$\hat{\mathbf{y}}' = C \times_1 \mathbf{U}_1 \times_2 \mathbf{q}'_2{}^T \mathbf{U}_2 \times_3 \mathbf{q}'_3{}^T \mathbf{U}_3 \times_4 \mathbf{q}'_4{}^T \mathbf{U}_4. \quad (2)$$

This expression can be further simplified by defining $\mathbf{q}_i^T \equiv \mathbf{q}'_i{}^T \mathbf{U}_i$ and analogously $\hat{\mathbf{y}} = \mathbf{U}_1^T \hat{\mathbf{y}}'$. Now applying $\times_1 \mathbf{U}_1^T$ to both sides of (2) and recalling that the columns the respective \mathbf{U} matrices are orthonormal we can write a more compact model representation as

$$\hat{\mathbf{y}} = C \times_2 \mathbf{q}_2^T \times_3 \mathbf{q}_3^T \times_4 \mathbf{q}_4^T, \quad (3)$$

where the unprimed coordinates refer to the latent code in the eigenspace spanned by the columns of the \mathbf{U}_i matrices. In this formulation, we have 3 individual parameter vectors and use repeated n -mode products to relate these to the model prediction.

We can rewrite (3) in a more general form to illustrate the mathematical structure of our model. Let us define the $P \times E \times R$, rank-1 parameter tensor $Q = \mathbf{q}_2^T \otimes \mathbf{q}_3^T \otimes \mathbf{q}_4^T$, where \otimes refers to the tensor product. Then the components of the rank-1 parameter tensor $Q \in \mathbb{R}^{P \times E \times R}$ is given by $Q_{\nu\rho\lambda} = q_\nu^{(2)} q_\rho^{(3)} q_\lambda^{(4)}$ where $q_\nu^{(k)}$ refers to the ν th component of the subspace vector $\mathbf{q}_k \in \mathcal{Q}_k$ for $k = \{2, 3, 4\}$.

With this definition, we can write (3) in a more compact and convenient representation using the Einstein summation convention

$$\hat{Y}^\mu = C^{\mu\nu\rho\lambda} Q_{\nu\rho\lambda}. \quad (4)$$

This lets us write the latent code, in the auxiliary latent space \mathcal{W} , as an application of the multilinear map, defined by the core tensor C , on the parameter tensor Q .

Our entire tensor model τ can thus be written as the composite map of the core C followed by the change-of-basis transformation defined by $\mathbf{U}_1 : \mathcal{W} \rightarrow \mathcal{W}$, and the inverse standardization operator $\Omega^{-1} : \mathcal{W} \rightarrow \mathcal{W}$, where Ω^{-1} translates and scales a latent vector back to the original scale of \mathcal{W} space according to the mean and variance of the BU-3DFE data.

C. Stacked Style-Separated Model

In addition to the previously presented model, we propose an alternative approach, where styles are separated instead of vectorizing the latent code. That is, we interpret the S styles of \mathbf{w} as separate vectors of dimension L , which is also indicated in Fig. 2. To separate the S styles, we

propose to order the latent codes into the data tensor $T_{\text{style}} \in \mathbb{R}^{S \times L \times P \times E \times R}$.

Then the shape dimension can be addressed separately by defining the style-specific tensors

$$T_s \in \mathbb{R}^{L \times P \times E \times R}, \quad s = 1, 2, \dots, S. \quad (5)$$

We factorize each style-specific tensor T_s , and define style-specific tensor model τ_s . The ensemble of these models is referred to as the *stacked style-separated model* τ_S , which has $S(P + E + R)$ parameters. In conclusion, while the prior vectorized model τ , based on T , has $P + E + R$ parameters, this formulation τ_S has $S(P + E + R)$ parameters since it models the style separately.

D. Optimization

Our next aim is to estimate the model parameters by constructing the estimator $\tau^{-1} : \mathcal{W} \rightarrow \mathcal{Q}$. The estimator is defined as the solution to the optimization problem

$$\begin{aligned} \min_{\mathcal{Q}} \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2 \quad \text{subject to} \\ \|\mathbf{q}_i\|_2^2 \leq c_2 \quad \text{and} \\ \|\mathbf{U}_i \mathbf{q}_i\|_1 \leq c_1 \quad \text{for } i = 2, 3, 4. \end{aligned} \quad (6)$$

The form of (6) is inspired by [11], [12], and enforces constraints on the model parameters to retrieve a stable representation of new latent vectors by linear combinations within the training data. We regularize the model using Ridge and Lasso regression. Then the Lagrangian for the constrained problem (6) can be written as

$$\begin{aligned} \mathcal{L}(Q, \lambda_1, \lambda_2) = \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2 \\ + \sum_{k=2}^4 \lambda_{2,k} \|\mathbf{q}_k\|_2^2 + \lambda_{1,k} \|\mathbf{q}'_k\|_1 \end{aligned} \quad (7)$$

where $\lambda_{1,k}, \lambda_{2,k} \geq 0$ refer to regularization parameters, i.e. Lasso and Ridge. Note that there is no prime on the Ridge term since $\|\mathbf{q}'_i\|_2^2 = (\mathbf{U}_i^T \mathbf{q}_i)^T (\mathbf{U}_i^T \mathbf{q}_i) = \|\mathbf{q}_i\|_2^2$ since $\mathbf{U}_i^T \mathbf{U}_i = \mathbf{I}$. We will now continue to present a strategy for solving the constrained optimization problem in (6) by Alternating Least Squares.

As in [11], [12] the minimization can be solved by first rewriting (3) as a matrix-vector multiplication separately for each of the three model parameter vectors as

$$\hat{\mathbf{y}} = \mathbf{A}^{(k)} \mathbf{q}_k, \quad k = 2, 3, 4, \quad (8)$$

where the matrices $\mathbf{A}^{(k)}$ are given by

$$\mathbf{A}^{(2)} = C \times_3 \mathbf{q}_3^T \times_4 \mathbf{q}_4^T, \quad (9)$$

$$\mathbf{A}^{(3)} = C \times_2 \mathbf{q}_2^T \times_4 \mathbf{q}_4^T, \quad (10)$$

$$\mathbf{A}^{(4)} = C \times_2 \mathbf{q}_2^T \times_3 \mathbf{q}_3^T. \quad (11)$$

Therefore, an unknown latent vector \mathbf{y} can be estimated by alternating between the systems (8), while updating the matrices $\mathbf{A}^{(k)}$ in each step.

IV. EXPERIMENTS

In the following, we give some additional details for the BU-3DFE database and continue to report on our experimental results.

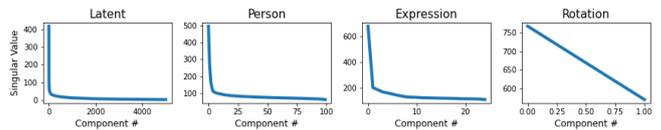


Fig. 5: Validation of decomposition results. Energy of singular values for each mode of T .

A. Facial Expression Database

As mentioned in the introduction, we use the BU-3DFE database [33]. The database contains 3D face scans and images of 100 persons (56 female and 44 male), with varying ages (18-70 years) and diverse ethnic/racial ancestries. Each subject was asked to perform the six basic emotions: anger, disgust, happiness, fear, sadness, and surprise, each with four levels of intensity. Additionally, for each participant, the neutral face was recorded. Hence, for each person, there are a total of 25 facial expressions recorded from two pose directions, left and right, resulting in 5000 face images.

B. Data Preprocessing

As a pre-processing step, we embedded each face image from the BU-3DFE database, into the latent space of StyleGAN, as described in Sec. II-B. We then collected the resulting latent vectors into the 4-way data tensor $T_0 \in \mathbb{R}^{N \times P \times E \times R}$. We then calculated the mode-1 unfolding $\mathbf{T}_0^{(1)} \in \mathbb{R}^{N \times PER}$ of T_0 containing all the PER latent vectors. We then standardized this matrix to zero mean and unit variance for each latent variable and then finally folded this standardized matrix into a $N \times P \times E \times R$ dimensional tensor T which we used for all subsequent experiments.

C. Subspace Analysis

The standardized tensor T was factorized by the HOSVD, as described in (1), yielding the four subspaces spanned by the columns of \mathbf{U}_k , $k = 1, \dots, 4$. The distribution of the energy of the subspaces is shown in Fig. 5, which illustrates the compactness of the subspaces.

In Fig. 6 we show a visualization of the expression subspace. As an initial step, we truncated the expression subspace from 25 dimensions to 3D. It can be seen that for each emotion, the variation in expression strength forms linear trajectories in expression space. These trajectories are star-shaped and meet at an origin of expression which is shared by all emotion trajectories. This is neither the neutral nor the mean face, but the ‘‘apathetic’’ face, found in [11], [12], see Fig. 7(a)-(c). In this case, the apathetic face in Fig. 7(c) is closer to the mean face than in [11], [12], displayed in Fig. 7(f) for comparison.

D. Vectorized vs. Stacked Style-Separated Model

In Sec. III we proposed to build two different versions of tensor models. (1) The *vectorized model* flattens each latent code of one image and then orders them into the tensor $T \in \mathbb{R}^{N \times P \times E \times R}$, and (2) the *stacked style-separated model* $T_{\text{style}} \in \mathbb{R}^{S \times L \times P \times E \times R}$ which considers the $S = 18$ styles of StyleGAN separately. We estimated the parameters for the

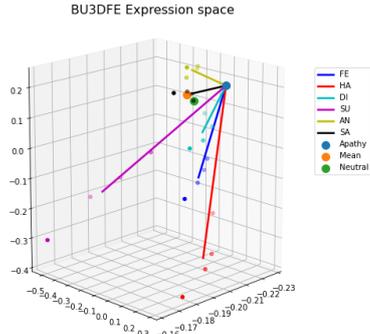


Fig. 6: Projection of the expression subspace, defined by \mathbf{U}_3 , onto 3 dimensions.

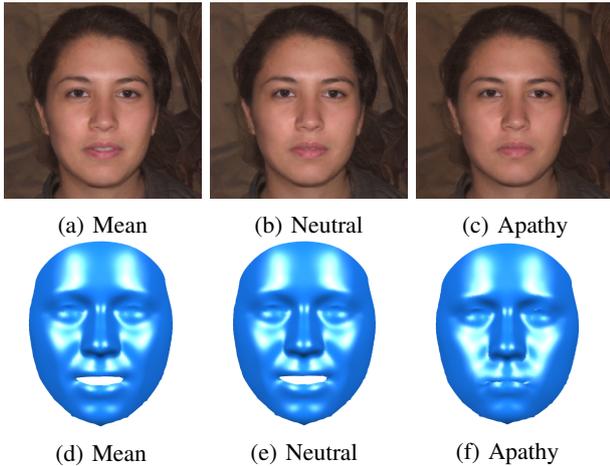


Fig. 7: Synthesized faces for (a) the mean face, (b) neutral face, and (c) apathetic face. Accordingly, (d), (e), (f) show the 3D faces synthesized by the method in [11].

two models, using the ALS procedure (8). The results are illustrated in Fig. 8. It can be seen, that the ground truth (Fig. 8a), is visually closer to the stacked style-separated model (Fig. 8c) than the vectorized model (Fig. 8b) for test images from the BU-3DFE data set (top row), as well as for arbitrary images (2nd and 3rd row). We conclude that the proposed adaptation by the separate styles improves performance.

E. Validation of Regularization Parameters

The optimization problem defined in (7) contains six regularization parameters $\lambda_{1,k}$ and $\lambda_{2,k}$, $k = 2, 3, 4$, two for each of the three parameter vectors, which must be manually set. In the following experiment we investigated how the hyperparameters influenced the quality of the results, and assume that they are the same for the three parameters, hence $\lambda_1 = \lambda_{1,k}$, and $\lambda_2 = \lambda_{2,k}$. Here we used the vectorized model on the basis of the standardized latent codes in (3). Initially, we divided the data into a training, validation, and test set by a randomized 90–5–5 split over the $P = 100$ person identities. The validation set thus had a total of $5ER = 250$ samples. We estimated the tensor model based

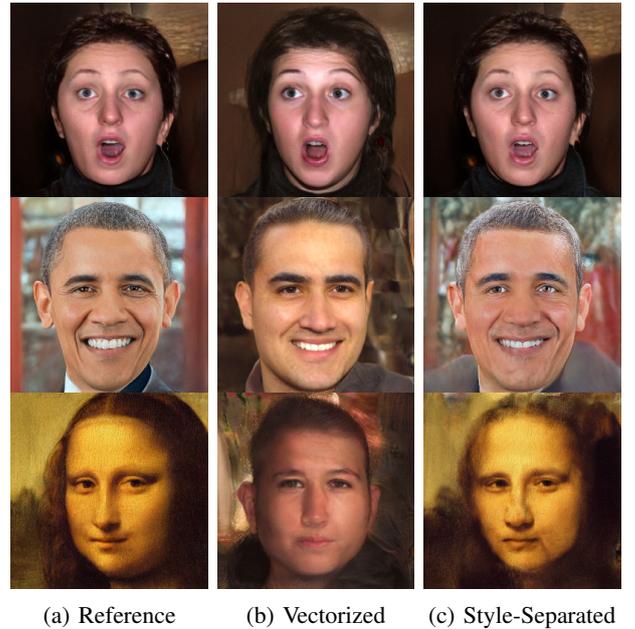


Fig. 8: Reconstructions: (a) Ground truth images, and the results from either (b) the vectorized model, and (c) the style-separated model. The top row shows an example from the BU-3DFE database, while the 2nd and 3rd rows illustrate reconstruction of novel images which are not part of BU-3DFE.

on the training set. For each latent vector in the validation set we then estimated the subspace parameters \mathbf{q}_i by ALS using (8).

We evaluated three kinds of errors for the validation set: the approximation error, and the expression and rotation transfer errors. The approximation error between the ground truth \mathbf{y} and estimated latent code $\hat{\mathbf{y}}_i$ is defined as $\epsilon_{\text{approx}} = \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2$. The transfer errors result from exchanging estimated parameters $\hat{\mathbf{q}}_k$ by known values $\tilde{\mathbf{q}}_k$. Hence using $\tilde{\mathbf{y}}_{\text{expr}} \equiv \tau(\hat{\mathbf{q}}_{\text{person}} \otimes \tilde{\mathbf{q}}_{\text{expr}} \otimes \hat{\mathbf{q}}_{\text{rot}})$ gives rise to an expression transfer error which we define as $\epsilon_{\text{expr}} = \|\tilde{\mathbf{y}}_{\text{expr}} - \mathbf{y}\|_2^2$. Analogously, the rotation transfer error is defined as the error arising from only changing the parameters associated with the rotation subspace according to $\epsilon_{\text{rot}} = \|\tilde{\mathbf{y}}_{\text{rot}} - \mathbf{y}\|_2^2$. The three error metrics ϵ_{approx} , ϵ_{expr} , and ϵ_{rot} were then calculated for each sample, with varying hyperparameter values λ_1 and λ_2 . In this experiment, we investigate Lasso and Ridge regression independently, i.e., we set $\lambda_1 = 0$ while varying λ_2 , and vice versa. We restrict ourselves to only consider cases where the regularization strength is equal for all subspaces.

The results are illustrated in Fig. 9. In general, it can be seen that the approximation error is more stable than the other two errors. Fig. 9a suggests that high values of λ_1 should be chosen for rotation transfer, while for expression transfer $\lambda_1 \approx 1$ seems to be a reasonable choice. Fig. 9b reveals that for $\lambda_2 \approx 1$ all error metrics are small, and hence this interval is a good choice.

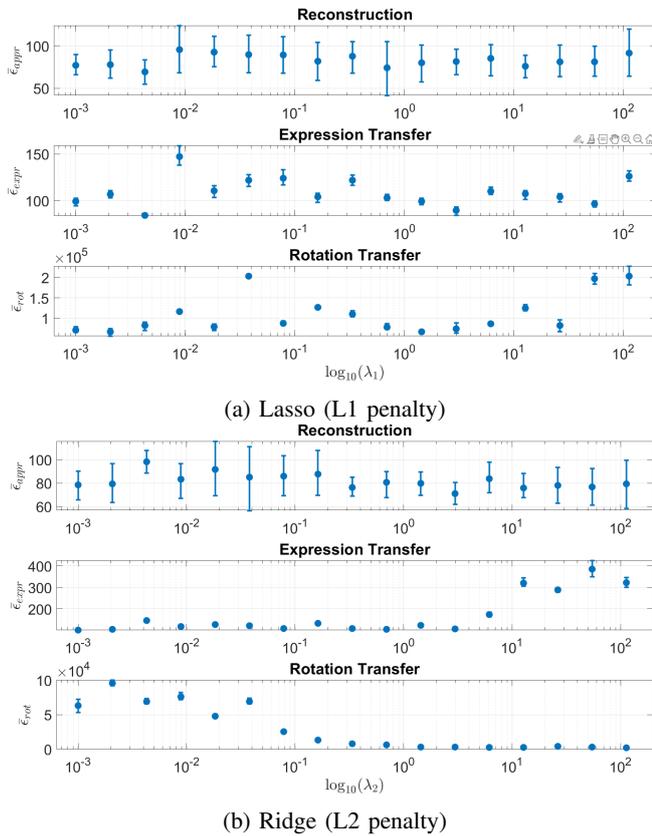


Fig. 9: Influence of the hyper parameters, λ_1 and λ_2 steering the (a) Lasso and (b) Ridge constraints, on (from top to bottom row) the approximation error, expression transfer error, and rotation transfer error.

F. Regularization and Parameter Transfer

We used the regularization parameters above to perform expression and rotation transfer on samples from the test set. We then synthesized images from the estimated parameters by applying the composite transformation $\hat{\mathbf{x}} = g(\tau(\hat{Q}))$ to the estimated subspace parameters \hat{Q} . Additionally, we performed expression and rotation transfer by replacing one of the three estimated parameter vectors by known values, as described before. We did this for the regularized model ($\lambda_1 > 0$, $\lambda_2 > 0$) and the non-regularized model ($\lambda_1 = \lambda_2 = 0$). Fig. 10 shows how well the ground truth, in \mathcal{W} space, (Fig. 10a) can be approximated by the non-regularized solution (Fig. 10b) and the regularized solution (Fig. 10c). It seems that the non-regularized solution matched the ground truth slightly better with respect approximation expression transfer. However, for rotation transfer (Fig. 10e) the regularized solution clearly outperformed the non-regularized solution. Because in the non-regularized solution the resulting image is not recognizable as a face anymore at all, while the regularized solution is not deformed and the rotation of the depicted faces conform to ground truth. This experiment thus showed that adding a small L2 regularization term yields stable rotation transfer.

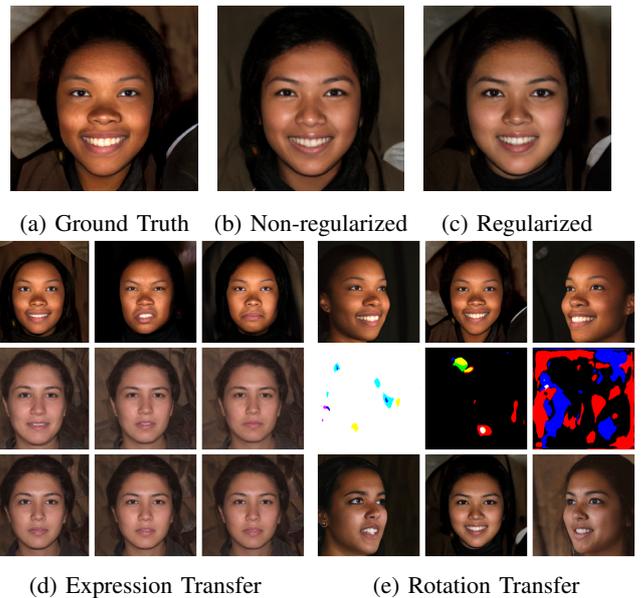


Fig. 10: Reconstruction and regularization results. (a) Ground truth (b) approximation by the non-regularized model, and (c) the regularized model. (d,e) Results from rotation and expression transfer containing ground truth (top row), the non-regularized solutions (middle row), and the regularized solution (bottom row).

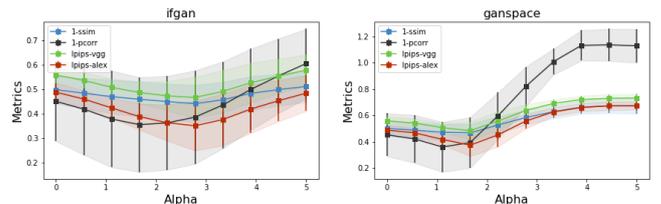


Fig. 11: To find the optimal interpolation strength α for rotation transfer for InterFaceGAN [24] and GANSpace [15] we compare the images generated by shifting the latent code corresponding to an image from the one rotation towards the other and compare the result with the ground truth.

G. Quantitative Comparison

Finally, we compare τ GAN to InterFaceGAN [24] and GANSpace [15] for the application of semantic face editing by using rotation transfer as one example.

Since the BU-3DFE database [33], see Sec. IV-A, contains 5000 faces images, 2500 from the left and from the 2500 right; we chose one of the two views as the reference image, and then used InterFaceGAN, GANSpace and τ GAN to estimate a reconstruction of the image from the complementary rotation. The resulting image was then compared to the Ground Truth (GT) by 1) Pearson correlation coefficient (pcorr), 2) Structural Similarity Index Measure (SSIM) [31], and 3) Learned Perceptual Image Patch Similarity (LPIPS) [34]. For the LPIPS measure, we employed two versions: one based on VGG [26], referred to as lpips-vgg, and the other, lpips-alex, on AlexNet [20].

In InterFaceGAN [24] the authors find semantic directions

of StyleGAN by fitting SVMs to single semantic attributes using an annotated data set. Using these directions, semantic editing can be performed by interpolating in the direction $\mathbf{n} \in \mathbb{R}^N$ defined by the SVM hyper-plane normal vector for a given latent code $\mathbf{w} \in \mathcal{W}$, as

$$\mathbf{w}_{\text{edit}} = \mathbf{w} + \alpha \mathbf{n}, \quad (12)$$

where α is the strength of the shift in semantic direction associated with \mathbf{n} . To perform rotation transfer, we chose the pose direction for the StyleGAN1 model trained on FFHQ provided by [24] as \mathbf{n} .

GANSpace finds semantic directions in an unsupervised fashion using PCA. The semantic meaning of the found principal components needs to be assigned by a one-time manual labeling. In the paper the authors report that the 10th principal component applied only to the first 7 layers produces a shift in rotation for the pretrained StyleGAN1 network. Using this definition, and the rotation direction, we can perform semantic edits with GANSpace in a similar way as in eq. 12.

To determine the optimal interpolation strength α for both methods, we design an experiment where we perform rotation transfer with varying values for α . From the latent code representing an image of one rotation, we edit the latent code towards the complementary rotation resulting in a latent vector \mathbf{w}_{edit} which is then used to synthesize an edited image. We then compare the edited image to the ground truth using the four metrics mentioned above. For each value of α we average the metrics and pick the minimum. The results are presented in Fig. 11, where it can be seen that the best performance for InterFaceGAN is reached at $\alpha = 2.77$, and for GANSpace at $\alpha = 1.66$, respectively. These values are used for the quantitative comparison presented in Fig. 13.

To perform rotation transfer with τ GAN model, we first estimated the model parameter vectors $\hat{\mathbf{q}}_k$, $k = 2, 3, 4$ for a given input image as described in Sec. III-D. Then we used the rotation subspace defined by \mathbf{U}_4 in (1). For τ GAN we take the subspace direction $\mathbf{m} = \mathbf{u}_2^{(4)} - \mathbf{u}_1^{(4)} \in \mathcal{Q}_R$, where $\mathbf{u}_1^{(4)}$, $\mathbf{u}_2^{(4)}$ are the first and second row of \mathbf{U}_4 , respectively. The rotation parameter was then changed as

$$\tilde{\mathbf{q}}_4 = \hat{\mathbf{q}}_4 + \gamma \mathbf{m}, \quad (13)$$

which then yields the edited latent code

$$\mathbf{w}_{\tau, \text{edit}} = \tau(\hat{\mathbf{q}}_2 \otimes \hat{\mathbf{q}}_3 \otimes \tilde{\mathbf{q}}_4). \quad (14)$$

Fig. 12 shows synthesized images produced by InterFaceGAN, GANSpace and τ GAN, respectively. These are compared against the reconstructions generated by latent codes interpolated directly in \mathcal{W} space by $\mathbf{w} = \beta \mathbf{w}_{\text{left}} + (1 - \beta) \mathbf{w}_{\text{right}}$ where \mathbf{w}_{left} and $\mathbf{w}_{\text{right}}$ refer to the left and right rotation, respectively. The results show that τ GAN provides an alternative way for generating rotation in the StyleGAN latent space. Compared to InterFaceGAN, our model seems to create rotations which better preserve features like skin tone and gaze direction, and compared to GANSpace the face shape seems better preserved. However, for all methods

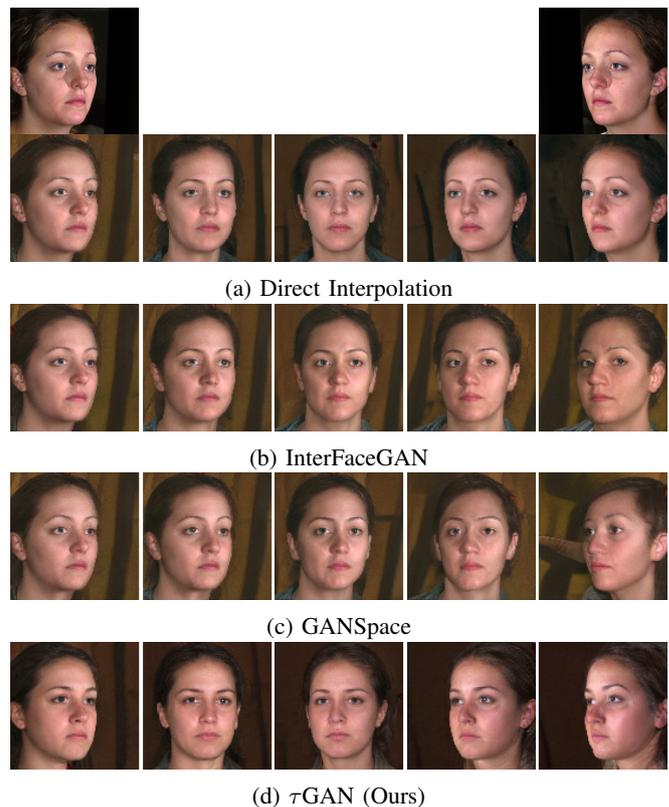


Fig. 12: Comparison of rotation transfer among varying methods. The ground truth images in pixel space are shown in the top row in the outermost columns. We use the latent code corresponding to the left hand rotation (top left) and try to recover the right hand rotation (top right). The provided images have been created by: (a) direct interpolation, (b) InterFaceGAN, (c) GANSpace, and (d) our proposed τ GAN.

we note that the identity of the person slightly changes in this example.

Additionally, we objectively compare the quality of rotation transfer resulting from different methods as follows. We apply the previously introduced three methods: InterFaceGAN, GANSpace, and our proposed τ GAN, to shift the rotation of the 125 left-oriented images in the validation set towards the right orientation. We then compare the edited images to the known ground truth using the same four metrics introduced at the beginning of this section. The results in Fig. 13 show that τ GAN has the lowest median value for all metrics when compared with InterFaceGAN and GANSpace.

V. CONCLUSIONS

In this work, we proposed τ GAN, a tensor-based model for the auxiliary latent space of the StyleGAN. It is constructed by first embedding the images of the BU-3DFE database into the latent space of StyleGAN. The latent codes were stored into a tensor which is then factorized into semantically meaningful subspaces by HOSVD. This construction ensured that the semantic directions were directly interpretable in contrast to unsupervised methods, where this

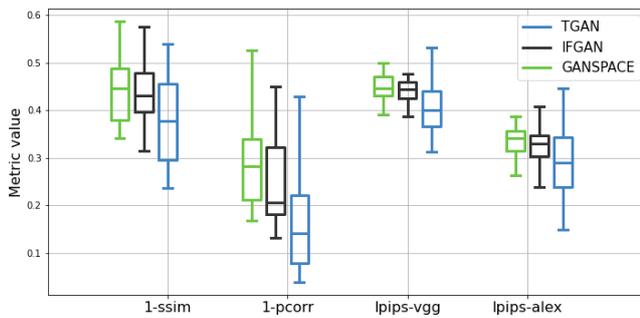


Fig. 13: Quantitative comparison of rotation transfer performed by varying methods. We start with images from the left rotation and shift the latent codes towards the right rotations using τ GAN, InterFaceGAN, and GANSpace. The edited images are then compared to the GT based on the previously used adapted metrics, redefined to be the lower the better. We observe that the edited images produced by τ GAN are more similar to the GT across all four metrics.

is not always the case.

We were able to generalize previous results [11] of face analysis by showing that the expression subspace has the structure where the expression trajectories meet in a specific *apathetic* expression, which is different from the mean or neutral face. We evaluated our approach quantitatively and qualitatively, and compared different versions of the proposed tensor models on the basis of approximation of unseen samples, and demonstrated the stability in the transfer of expression and rotation. From the results, we conclude that the proposed approach is a powerful way for characterizing and parameterizing the latent space of StyleGAN.

The current setting assumes complete data that contains measurements of all the people performing the same expressions from each rotation without any missing data. This requirement could be relaxed by low-rank completion methods that is left for future work. To conclude we employed a model trained on FFHQ, and received promising results on the BU-3DFE data set.

REFERENCES

- [1] R. Abdal, Y. Qin, and P. Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proc. ICCV*, pages 4431–4440, 2019.
- [2] P. Baylies. Stylegan encoder - converts real images to latent space. <https://github.com/pbaylies/stylegan-encoder/>, 2019.
- [3] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proc. SIGGRAPH*, pages 187–194, 1999.
- [4] A. Brunton, T. Bolkart, and S. Wuhler. Multilinear Wavelets: A Statistical Shape Space for Human Faces. In *Proc. ECCV*, pages 297–312, Jan. 2014.
- [5] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou. FaceWarehouse: A 3D Facial Expression Database for Visual Computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, Mar. 2014.
- [6] X. Chen, Y. Duan, R. Houthoofd, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, editors, *NIPS*, pages 2172–2180, 2016.
- [7] L. De Lathauwer and B. De Moor. A multi-linear singular value decomposition. *Society for Industrial and Applied Mathematics*, 21:1253–1278, 03 2000.

- [8] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong. Disentangled and controllable face image generation via 3d imitative-contrastive learning. In *IEEE Computer Vision and Pattern Recognition*, 2020.
- [9] C. Ferrari, G. Lisanti, S. Berretti, and A. D. Bimbo. A dictionary learning-based 3d morphable shape model. *IEEE Transactions on Multimedia*, 19(12):2666–2679, 2017.
- [10] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS’14, page 2672–2680, Cambridge, MA, USA, 2014. MIT Press.
- [11] S. Graßhof, H. Ackermann, S. Brandt, and J. Ostermann. Apathy is the root of all expressions. *12th IEEE Conference on Automatic Face and Gesture Recognition (FG2017)*, 2017.
- [12] S. Graßhof, H. Ackermann, S. S. Brandt, and J. Ostermann. Multilinear Modelling of Faces and Expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3540–3554, Oct. 2021. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Proc. CVPR*, pages 770–778, 2016.
- [14] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proc. ICCV*, 2017.
- [15] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris. Ganspace: Discovering interpretable gan controls. In *Proc. NeurIPS*, 2020.
- [16] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila. Training generative adversarial networks with limited data. In *Proc. NeurIPS*, 2020.
- [17] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proc. CVPR*, pages 4396–4405, 2019.
- [18] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR*, 2020.
- [19] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM REVIEW*, 51(3):455–500, 2009.
- [20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS’12, page 1097–1105. Curran Associates Inc., 2012.
- [21] D. Nikitko. Stylegan – encoder for official tensorflow implementation. <https://github.com/puzer/stylegan-encoder/>, 2019.
- [22] D. Y. Park and K. H. Lee. Arbitrary style transfer with style-attentional networks. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019-June:5873–5881, Dec 2018.
- [23] S. Pidhorskyi, D. A. Adjeroh, and G. Doretto. Adversarial latent autoencoders. In *Proc. CVPR*, June 2020.
- [24] Y. Shen, J. Gu, X. Tang, and B. Zhou. Interpreting the latent space of gans for semantic face editing. In *CVPR*, 2020.
- [25] Y. Shen and B. Zhou. Closed-form factorization of latent semantics in gans. In *CVPR*, 2021.
- [26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.
- [27] A. Tewari, M. Elgharib, G. Bharaj, F. Bernard, H.-P. Seidel, P. Pérez, M. Zöllhofer, and C. Theobalt. Stylerig: Rigging stylegan for 3d control over portrait images, cvpr 2020. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2020.
- [28] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *Proceedings of the 7th European Conference on Computer Vision-Part I*, ECCV ’02, page 447–460, Berlin, Heidelberg, 2002. Springer-Verlag.
- [29] M. A. O. Vasilescu and D. Terzopoulos. Multilinear subspace analysis of image ensembles. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages II– 93, 07 2003.
- [30] D. Vlastic, M. Brand, H. Pfister, and J. Popović. Face Transfer with Multilinear Models. In *ACM SIGGRAPH*, pages 426–433, 2005.
- [31] Z. Wang and A. Bovik. A universal image quality index. *Signal Processing Letters, IEEE*, 9:81 – 84, 04 2002.
- [32] K. Yano and K. Harada. Multilinear face model. In *Visualization, Imaging, and Image Processing (VIIP 2008)*, 2008.
- [33] L. Yin, X. Wei, Y. Sun, J. Wang, and M. Rosato. A 3d facial expression database for facial behavior research. In *7th Intern. Conf. on Automatic Face and Gesture Recognition (FG06)*, pages 211–216, 2006.
- [34] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.