Learning to Generate 3D Stylized Character Expressions from Humans

Deepali Aneja[1], Bindita Chaudhuri[1], Alex Colburn[2], Gary Faigin[3], Linda Shapiro[1] and Barbara Mones[1]

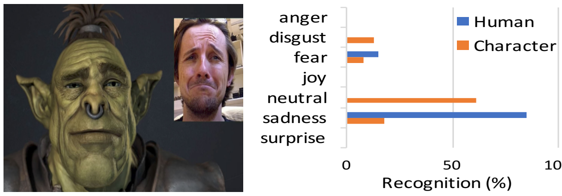
[1]Paul G. Allen School of Computer Science & Engineering, UW, Seattle, WA, USA
[2]Zillow Group, Seattle, WA, USA
[3]Gage Academy of Art, Seattle, WA, USA
{deepalia,bindita,alexco,shapiro,mones}@cs.washington.edu, gary@gageacademy.org

Abstract

*We present **ExprGen**, a system to automatically generate 3D stylized character expressions from humans in a perceptually valid and geometrically consistent manner. Our multi-stage deep learning system utilizes the latent variables of human and character expression recognition convolutional neural networks to control a 3D animated character rig. This end-to-end system takes images of human faces and generates the character rig parameters that best match the human's facial expression. ExprGen generalizes to multiple characters, and allows expression transfer between characters in a semi-supervised manner. Qualitative and quantitative evaluation of our method based on Mechanical Turk tests show the high perceptual accuracy of our expression transfer results.*

1. Introduction

Our work is motivated by the goal of enhancing animated storytelling by improving 3D stylized character facial expressions. The importance of believable and accurate animated character facial expressions is readily demonstrated by films and games such as Polar Express [47] and Mass Effect: Andromeda [2]. In these examples, it is difficult for the audience to connect to the characters and broader storyline, because the characters do not exhibit clearly recognizable facial expressions that are consistent with their emotional state in the storyline [39, 34]. Characters must have *perceptually valid* expressions, that are clearly perceived by humans to be in the intended expression class. Fig. 1 shows a concrete example of a perceptually invalid expression, in which the human expression did not transfer correctly to the character when tested on Mechanical Turk (MT) for expression clarity with 30 test subjects.



(a)

(b)

Figure 1: Example of inaccurate expression transfer. (a) Expression transfer from human (top right) to a character [17]. (b) Mechanical Turk testers perceive the human expression as sadness, while the character expression is perceived as neutral and a mixture of other expressions. The character expression has neither expression clarity nor geometric consistency.

Animator-created character expressions can be expressive and clear but require expertise and hours of work. In order to speed up the animation process, animators often use human actors to control and animate a 3D stylized character using a facial performance capture system. These systems often lack the expressive quality and perceptual validity of animator-created animations, mainly due to their assumption that geometric markers are sufficient for expression transfer. The geometry-based methods and retargeting [26] based on handcrafted descriptors may be unable to take into account the perception of the intended expression when transferred onto a stylized character. We are unaware of any tools or methods that support animators by validating the perception of character expressions during creation. Despite recent advances in modeling capabilities, motion capture and control parameterization, current methods do not address the fundamental problem of creating clear expressions that humans recognize as the intended expression.

Our goal is to learn 3D stylized character expressions from humans in a *perceptually valid* and *geometrically consistent* manner. To this end, we propose an end-to-end system, ExprGen, that takes a 2D image of a human and predicts the 3D rig parameters of a character. This is a challenging goal because there is no existing dataset mapping

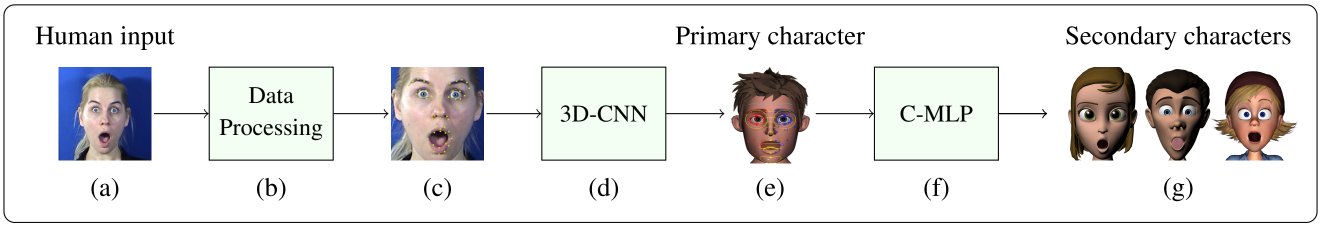


Figure 2: Overview of our multi-stage expression transfer system ExprGen: (a) 2D images of human facial expressions are pre-processed (b, c, Sec. 3). A CNN (d, Sec. 4.2.1) generates rig parameters corresponding to the human expression for primary characters (e). A separate neural network (f, Sec. 4.2.2) performs primary-character-to-secondary-character expression transfer (g).

2D images of human expressions to 3D character rig parameters. Further, it is prohibitively expensive to manually create a dataset with explicit human image to 3D rig parameter labeling. To address this challenge, we leverage publicly available human and character expression datasets with 2D images and expression labels [12, 24, 33, 25, 28, 3]. Each 2D character image in the dataset is rendered from its 3D facial rig (which has associated rig control parameters). In our work, the system learns from the six broad categories (anger, disgust, fear, joy, sadness, surprise) [15] and neutral, since there is agreement on their recognition within the facial expression research community, and these seven expressions occur in a wide range of intensities and can blend with each other to create additional expressions.

Our approach is to learn the correspondence between 2D images of human and character expressions and use this correspondence to map human expression images to 3D character rig parameters. We start with recognizing facial expressions for both humans and characters to induce a perceptual metric and constraints for expression generation and matching. Our system then learns a joint embedding to map human expressions to character expressions. This mechanism also accounts for geometric consistency, so that any mapping is both perceptually valid and geometrically consistent. At this point, we can take any human expression image and find similar or dissimilar images in a database of character images. Using this similarity analysis, we train a regression network, 3D-CNN, from a human expression onto the parameters of a specific or *primary* character 3D rig. Finally, a lightweight mechanism, *Character Multi-Layer Perceptron* (C-MLP), transfers character expressions to other characters. This enables re-use of a primary character rig trained in the previous steps to drive *secondary* characters. Fig. 2 depicts an overview of our system at run time. Images of human facial expressions are initially processed to detect the face and 49 geometric landmarks, and then fed into the 3D-CNN to generate expression specific parameters for the primary character rig. The C-MLP uses the generated expression parameters to produce the expression on other secondary 3D stylized characters.

The main contributions of our work are: (1) A novel perceptually valid method to map 2D human face images to 3D stylized character rig controls. (2) The ability to utilize this mapping to generate 3D characters with clear unambiguous facial expressions. (3) A semi-supervised method to enable expression transfer between multiple characters.

ExprGen uses images of human facial expressions to generate perceptually valid and geometrically consistent 3D stylized character expressions using a deep learning framework. Both qualitative and quantitative results detailed in Sec. 5 illustrate the accurate, plausible, and perceptually valid expression transfer from humans to 3D stylized characters. Our approach is unique in that it uses geometry plus perceptual validity rather than pure geometry-based mathematical operations. Our method also introduces a simple lightweight expression transfer mechanism between characters, which enables one character rig to drive any other rig and eliminates the need to individually train each rig with the full pipeline. We hope that our method will enable the creation of new perceptually driven tools that help animators create clear and accurate character expressions and ultimately successful animated stories.

2. Related Work

Facial Expression Recognition (FER). With recent advancement in deep learning research, Convolutional Neural Networks have shown great improvement in the FER tasks [30, 6, 23, 21, 13, 22] and there are a number of fusion algorithms to boost the recognition performance [42, 46, 38, 44]. Many systems are tuned on a single facial expression database, which makes them sensitive to the lighting and particular poses present in that database. These methods focus on engineered features, which lack the generalizability to perform "in the wild" expression recognition. To overcome this limitation, we combine human databases from different sources including a dataset collected in the wild for our training step in order to improve the robustness of our trained model. In our work, we use CNNs to learn perceptual features pertaining to facial expressions and combine them with geometric features, since this approach has been shown to perform better in expression recognition and transfer to stylized characters than geometric features alone [3]. Note that Aneja *et al.* [3] proposed a retrieval method to identify the closest 2D character expression image from the existing database to a given human image, whereas we propose a method that generates a 3D stylized character expression for a given human image.

Expression Transfer. Facial animation of stylized characters by retargeting human expressions can be classified into two main categories: 1) parametrization or generating facial animation parameters (based on geometry such as nose width, eye opening etc. [5] or physical control parameters such as muscle action [16]), and 2) motion retargeting [9], which involves mapping motion vectors directly to the target face model [31]. However, dense mesh motion required for motion retargeting may not be available from some input systems. Various other techniques like regression [35, 19], PCA-based linear modeling [11, 27] and blendshape mapping [37, 7, 4] have also been used to learn the mappings between an human and character faces. PCA-based models are insufficient to represent the particularly detailed variations of a human facial expressions due to their limited dimensionality. In contrast to PCA models, blendshapes have the important advantage that a given facial expression among different people corresponds to a similar set of basis weights. Instead of using a fixed parametric shape model or relying upon the expression specific blendshape weights, our deep-learning-based approach learns the perceptual and geometric features together from a database collected from various sources that can flexibly represent the particular facial expressions of the user without the need of depth information. A similar approach has been proposed in [44], but the method is limited to transferring expressions to 2D cutout animations only.

Generalization to multiple 3D characters. Visual storytelling involving multiple stylized characters necessitates facial expression manipulation of multiple characters with minimum human effort. Several marker-based and markerless facial motion capture software packages have been recently developed, including Faceshift Studio [17], Faceware Analyzer and Retargeter [18], Mixamo Face Plus [29], Dynamixyz Performer Suite [14], and Optitrack Expression [32]. The traditional marker-based products create some predefined marker points on the human face and map them to the corresponding points on the 3D character rigs, enabling live tracking of the human facial motion and character facial animation. However, the limited number of marker positions often fail to capture the intended expression. The markerless systems like Faceshift and Faceware capture the blendshapes associated with a set of standard expressions made by the human source and map the blendshapes onto stylized characters. However, all these methods require a significant amount of manual effort in terms of setting up a new character, mapping the expressions from the existing character to a new one, and refining the generated expressions for accurate tracking.

3. Data Acquisition and Processing

Our framework uses two databases: (1) Human expression database (HED) and (2) Character expression database (CED). The details about these databases are given below:

Human Expression Database (HED). We combine five publicly available labeled facial expression databases to create the HED: (a) Static Facial Expressions in the Wild (SFEW) database [12], (b) Extended Cohn-Kanade database (CK+) [24], (c) MMI database [33], (d) Karolinska Directed Emotional Faces (KDEF) [25], and (e) Denver Intensity of Spontaneous Facial Actions (DISFA) database [28]. The HED consists of approximately 100K labeled images; the number of samples for each class is balanced to avoid bias towards a particular expression. Specifically, we under-sampled the neutral class, so that its distribution is the same as the other expression classes.

Character Expression Database (CED). We use FERG-DB [3] which consists of 55,767 labeled face images of six stylized characters ('Mery', 'Aia', 'Bonnie', 'Jules', 'Malcolm' and 'Ray') for training. In addition to FERG-DB, we add three new characters ('Tuna' [41], 'Mathilda' and 'Cody' [1]) for validation. An animator created 10 key poses per expression for each new character and labeled them using Mechanical Turk (MT) with 70% agreement among 50 test subjects. We also obtained the stylized 3D rigs modeled using the Autodesk®MAYA software for all the characters and used the control parameters associated with them in our work.

To pre-process our human input as shown in Fig 2(a-c), we extract 49 facial landmarks [40] to register a face to an average frontal face via an affine transformation and use the landmarks to extract the geometric features including the following measurements: left/right eyebrow height (vertical distance between top of the eyebrow and center of the eye), left/right eyelid height (vertical distance between top of an eye and bottom of the eye), nose width (horizontal distance between leftmost and rightmost nose landmarks), mouth width (left mouth corner to right mouth corner distance), closed mouth measure (vertical distance between the upper and the lower lip), and left/right lip height (vertical distance between the lip corner from the lower eyelid). Each distance is normalized by the bounding box of the face. We extract the geometric features for the character images in the same manner. Once the faces are cropped and registered, the images are re-sized to 256 × 256 pixels for input to our network training.

4. Methodology

In order to build a system that can transfer human expressions to multiple 3D characters, we need several components to handle the human-to-character transfer in 2D,

Layer type	Filter size	Stride	Output
CONV-1	11x11	1	64x256x256
CONV-2	1x1	2	64x128x128
CONV-3	5x5	1	64x128x128
CONV-4	1x1	2	64x64x64
CONV-5	5x5	1	64x64x64
CONV-6	1x1	2	64x32x32
CONV-7	3x3	1	64x32x32
CONV-8	1x1	2	64x16x16
CONV-9	3x3	1	64x16x16
CONV-10	1x1	2	64x8x8
CONV-11	3x3	1	64x8x8
Avg. Pooling-12	8x8	1	64x1x1
FC-13			1x7
FC-14			1x7

Table 1: HCNN and SCNN network architecture

produce parameters for a primary character expression in 3D including both perceptual and geometric similarity, and transfer the expression of a primary character to multiple secondary characters. We build a multi-stage deep learning system ExprGen with two major components: Training from 2D Datasets (Sec. 4.1) and 3D Expression transfer. 3D Expression transfer is composed of two separate components: Human to Character transfer (Sec. 4.2.1) and Character to Character transfer (Sec. 4.2.2).

4.1. Training from 2D Datasets

The goal of this step is to learn a joint embedding between human and primary character expressions based on perception and geometry. Our approach is inspired by the recent success of CNNs to learn the image similarity based on Pseudo-Siamese networks [8, 45]. We extend this concept for expression similarity application by fusing the perceptual and geometric features of humans and characters. We train a Pseudo-Siamese network called the fused-CNN (f-CNN) with two branches, Human CNN (HCNN) and Shared CNN (SCNN). We first train the HCNN on the human expression dataset (HED) to classify an input human face image into one of the seven expression classes. Then, we initialize the weights of the SCNN with those of HCNN, except for the Fully Connected (FC) layers and train the SCNN on FERG-DB by transfer learning [43]. In this process, the last layer of the HCNN is fine-tuned with FERG-DB (for every character) by continuing the backpropagation learning step, creating a shared embedding feature space. The network structures for the HCNN and SCNN are given in Table 1. We did not find any significant improvement with more layers or higher dimensionality of fully-connected layers.

After the network branches are trained to recognize the expressions on humans and characters independently, we concatenate the outputs from their average pooling layers and send it to a network of two FC layers to form f-CNN as shown in Fig. 3. To train the f-CNN, we introduce a similarity measure based on the distance between two image encodings as follows. After the HCNN predicts the perceptual expression label of the human input image, the FERG-DB is searched to retrieve the character images having the same predicted label. Then, the Euclidean distance between the geometric feature vector of the human image and those of all the retrieved character images are computed and ordered based on the distance to the human image. Note that we did not always find perfect matches. However, our dataset is large enough to enable the CNNs to learn generalizable matching representations between human and character images. To solve the issue of incorrect geometry match within the same expression class, triplets $(E, E+, E-)$ of training images are created where:

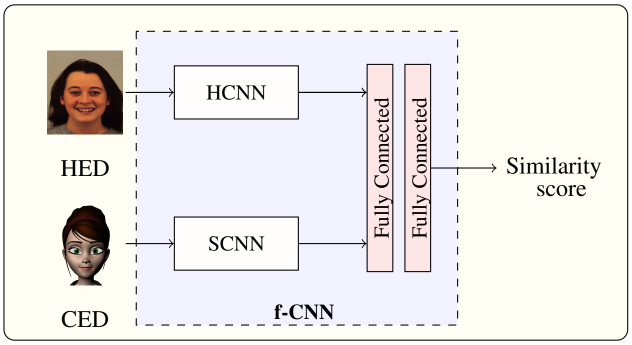


Figure 3: The HCNN and SCNN are fused together to form the f-CNN, which is trained to produce a similarity score between human and primary character expressions.

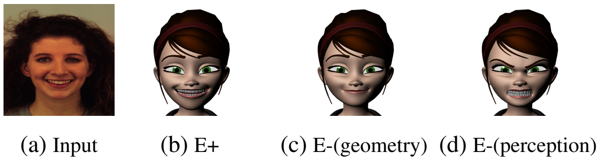

(a) Input (b) E+ (c) E-(geometry) (d) E-(perception)

Figure 4: Comparison of best matches for training the f-CNN based on geometry and perception. (a) Human input (E), (b) Positive match (E+), (c) Negative match with incorrect geometry (E-), (d) Negative match with incorrect perception (E-).

1. *E* is a reference human expression image.
2. *E+* is a character image similar to the reference human expression image (best geometry match in the search).
3. *E-* is another image that is not geometrically and/or perceptually similar to the reference human expression image. For example, if E is an open mouth joy human expression, then character anger retrieval would be incorrect perceptually and closed mouth human joy would be incorrect geometrically as shown in Fig. 4.

The f-CNN takes a human expression image and primary character expression image and produces a similarity score by minimizing a loss function consisting of a hinge-based loss term and a squared *L2*-norm regularization term [36]:

$$\min_w \sum_{i=1}^{N} \max(0, 1 - l_i y_i^{net}) + \frac{\lambda}{2} \|w\|_2 \qquad (1)$$

where y_i^{net} is the network output for the i^{th} training sample, $l_i \in \{-1, 1\}$ is the corresponding label (with +1 and -1 denoting a non-matching and a matching pair, respectively) and w are the weights of the neural network. The hinge loss minimizes the distance between E and E+ (matching both geometry and perception) and maximizes the distance between E and E- (mismatching the geometry and/or perception). Similar to the approach described in [36], triplets are generated online by selecting the hard positive/negative exemplars from within a mini-batch for our training. The softmax layer at the end of f-CNN converts the similarity score to binary classification (similar or dissimilar).

4.2. 3D Expression Transfer

This step generates perceptually valid 3D characters from images of human expressions. It is divided into two stages: expression transfer from human to a primary character rig and expression transfer from primary to secondary character rigs. The stages are described as follows:

4.2.1 Human to Character Transfer

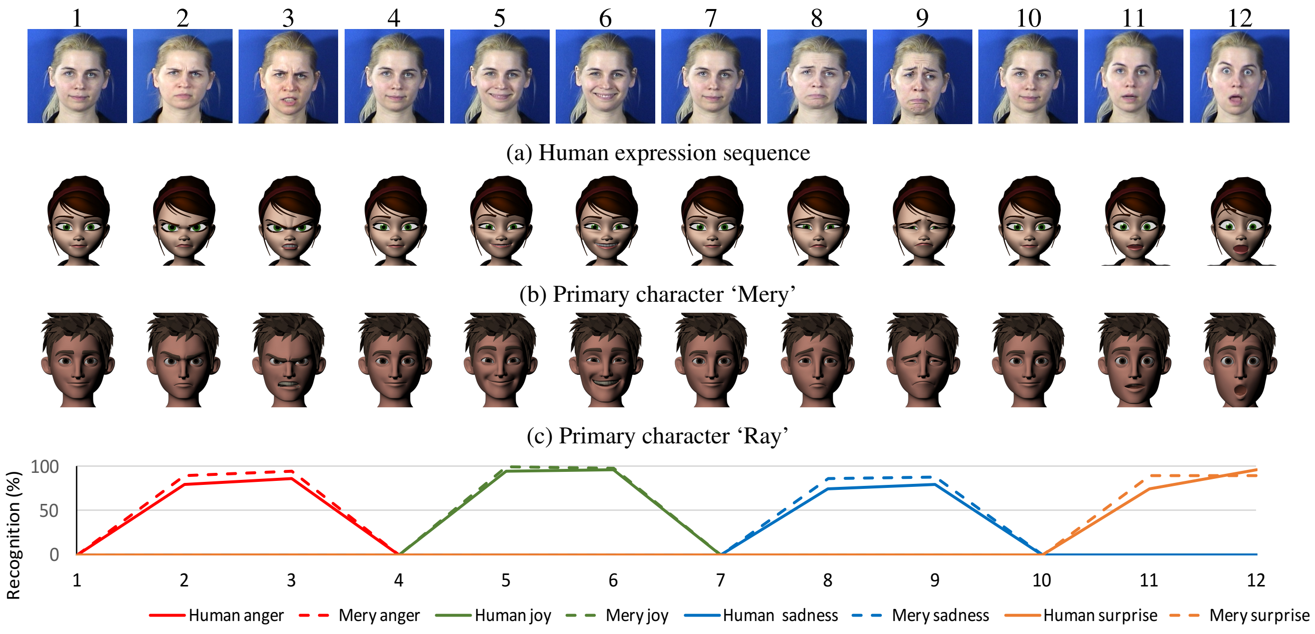
The f-CNN can be used to retrieve the matching 2D character expressions; however, it requires a database of character images. We aim to control the primary rig by producing rig parameters for any given human image. To control the rig in 3D, we train another CNN called the 3D-CNN which has the same configuration as shown in Table 1 except for the dimensionality of the FC layers. Instead of seven probabilities for classifying seven expression classes, the final output is the parameters for the primary character. We initialize the weights of the 3D-CNN by trained HCNN weights so that we can transfer the knowledge learned from the HED, and the model does not overfit the 3D-parameters dataset. The pairs of a human input image and the 3D-parameters corresponding to the 2D character image with similar expression (as obtained at the output of f-CNN) are used for training the 3D-CNN (Fig. 2(d)).

All the networks are trained end-to-end using the Torch framework [10] until convergence using stochastic gradient descent with hyper parameters (momentum of 0.9, weight decay of 0.0005 and a batch size of 50) on a single NVIDIA GTX-1080 GPU. In order to make sure that the pre-trained weights are not drastically changed, the learning rate for the SCNN, f-CNN and 3D-CNN is set lower (0.0001) than that of the HCNN (0.001). The learning rate was dropped by a factor of 10 after every 10 epochs of training. Batch normalization was applied [20] after every convolutional layer to reduce the internal-covariate-shift, ReLU as the activation function and drop out with the drop-out ratio of 0.2. To avoid overfitting, our training data is augmented by horizontal flipping, rotating, and random cropping followed by scaling. We used an 80:10:10 split for training, validation and test sets, and performed 5-fold cross validation.

4.2.2 Character to Character Transfer

ExprGen is trained for a primary character rig, and we propose a lightweight alternative to training a different network for each new secondary character as shown in Fig 2(e-g). Due to the absence of one-to-one correspondence between the facial control points on different rigs, manual mapping of the rig parameters is often not possible. Our character-to-character expression transfer model aims at automatically learning a function to map the 3D-parameters of the primary character to the secondary characters. For each secondary character we create a separate multilayer perceptron (MLP), which is a one-hidden-layer neural network with M input nodes, N output nodes and $\frac{1}{2}(M+N)$ hidden nodes with *tanh* activation, where M and N are the number of 3D-parameters of the primary and the secondary characters respectively. Gradient descent is used with a mini-batch size of 10 and a learning rate of 0.005 to minimize the square loss between the input and output parameters. These networks (together called C-MLP) are trained in parallel and then augmented at the end of the 3D-CNN to map the input human expression simultaneously on multiple stylized characters.

We obtained pairs of training examples for the C-MLP by using a combination of two distance measures: d_{geometry} and $d_{\text{perception}}$. $d_{\text{geometry}} = ||f_p^g - f_s^g||_2$ is the Euclidean distance between the geometric feature vectors of the primary (f_p^g) and secondary character (f_s^g) image pairs, while $d_{\text{perception}} = ||f_p^p - f_s^p||_2$ is the Euclidean distance between the perceptual feature vectors (f_p^p and f_s^p) of the image pairs. The perceptual features are obtained by extracting the output of the last FC layer of the SCNN and normalizing it by the softmax weight as done in [3]. Given a primary character with an expression to find on a secondary character in FERG-DB (on which our SCNN is trained), all secondary character images in the CED having the same perceptual label as the primary character image are retrieved and ordered by the smallest value of d_{geometry}; the image with smallest distance value is returned. If the secondary character is not in FERG-DB, based on empirical evidence, the images of the secondary character for the perceptual labels having the two highest probabilities are retrieved and the combined function $\frac{1}{2}(d_{\text{perception}} + d_{\text{geometry}})$ is used to order them for retrieval. This methodology produces a set of matching (primary character, secondary character) pairs, for which we have both images *and* the 3D parameters that can be used to generate the 3D meshes from which those



(a) Human expression sequence

(b) Primary character 'Mery'

(c) Primary character 'Ray'

(d) Plot showing percentage of 30 MT test subjects recognizing the correct expression on human and the character 'Mery'

Figure 5: Human to primary character expression transfer for human expression transition from neutral to joy, from neutral to surprise, from neutral to sadness, and from neutral to anger based on both perceptual and geometric similarity. (a) Human input expression frames (1-12), (b) Mapped expressions on 'Mery', and (c) Mapped expressions on 'Ray', (d) Expression recognition results between human (solid lines) and transferred expressions on 'Mery' (dashed lines) for different expressions.

images are derived. The pairs of corresponding parameters are used to train the C-MLP. Once trained, the C-MLP transforms the 3D parameters of a primary character into the corresponding 3D parameters of a secondary character.

5. Results

We evaluated the performance of our system by computing the expression recognition accuracies of the HCNN and SCNN independently, testing the human-to-character expression transfer perceptual accuracy and comparing our results with Faceware (commercial product). In all the subsequent figures and tables, we show the 2D rendered images of 3D character rigs and use the following notation for the expression classes - A: anger, D: disgust, F: fear, J: joy, N: neutral, Sa: sadness, Su: surprise.

5.1. Expression Recognition Accuracy

We first evaluated the HCNN and SCNN for the expression recognition task using the HED and CED in a 10-fold cross-validation setting. The HCNN and SCNN obtained average accuracies of 89.71% and 96.82%, respectively. We note that our classification networks perform better than the prior networks trained for a similar classification task [3] because of training the HCNN on an additional dataset to learn the features in the wild. The accuracy of our networks increased by about 5% when we did not apply the max pooling step after every convolution layer, indicating that average pooling after all the convolution layers helps the network to preserve the facial appearance and subtle distinctions between each expression, which is lost when max pooling is applied after every convolution. However, our focus is not on the classification accuracy of the trained networks, but on using them to produce 3D-rig parameters. In the remaining experiments we use these networks to learn the expression feature space for humans and stylized characters and use their weights to initialize our final 3D-CNN.

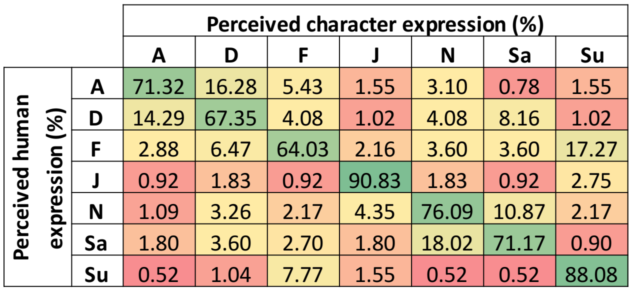
5.2. Human to Character Expression Transfer

To evaluate our results for clarity in expression recognition and perceptual accuracy of the transferred expression, we asked 30 MT test subjects to recognize the input human expression and the generated primary character expression (output of 3D-CNN) for 1000 expression transfer results (approx. 150 for each expression class) on different stylized characters.

We computed the clarity of expression recognition on human and characters independently by comparing the perceived expressions with the ground truth labels. The average expression recognition accuracies for humans and characters are shown in Table 2. We observe that the character expression recognition accuracies are higher than humans, since the characters have simpler geometry and stylization can make the expressions relatively easier to perceive. Surprise and joy show high accuracy, while disgust and fear are

Class	A	D	F	J	N	Sa	Su
Human	76.27	63.81	68.47	94.31	78.03	72.95	92.26
Character	90.45	72.89	79.16	96.39	84.38	79.44	94.87

Table 2: Average (%) expression recognition accuracy for 2D images of human and stylized character expressions when compared with the ground truth labels respectively. Note that the characters have higher expression clarity than humans due to their simpler geometry.



Perceived human expression (%)	Perceived character expression (%)						
	A	D	F	J	N	Sa	Su
A	71.32	16.28	5.43	1.55	3.10	0.78	1.55
D	14.29	67.35	4.08	1.02	4.08	8.16	1.02
F	2.88	6.47	64.03	2.16	3.60	3.60	17.27
J	0.92	1.83	0.92	90.83	1.83	0.92	2.75
N	1.09	3.26	2.17	4.35	76.09	10.87	2.17
Sa	1.80	3.60	2.70	1.80	18.02	71.17	0.90
Su	0.52	1.04	7.77	1.55	0.52	0.52	88.08

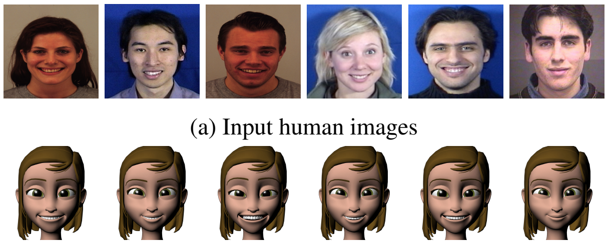
Figure 6: Confusion matrix for perceived transferred expression recognition (%) for seven expression classes.

more difficult for humans to both perceive and act out.

To test the accuracy of the expression transfer we compared human expressions to that of the generated primary character. We used the perceived label (as perceived by MT subjects) of the human as the ground truth and the perceived label of the character as the predicted output in its human-character-transfer pair. Fig. 6 shows the confusion matrix for transferred expression recognition for each expression class. For a given row (e.g. anger), the columns represent the percentage (averaged over all the perceived human anger expressions) of MT subjects agreeing on the corresponding expression classes for the transferred character expressions. The values show that ExprGen results in accurate transfer of expressions for most of the classes with an average correct perceptual recognition rate of 75.55%. The most common errors are confusion between disgust and anger, between fear and surprise, and between neutral and sadness. These errors are intuitively reasonable since the confused expressions have similar-looking geometric configurations. The least accurate expression transfer was for disgust and fear but as Table 2 shows, these expressions are difficult to recognize for both human and character images.

5.2.1 Single Human to Multiple Characters

ExprGen generates expressions for multiple characters with high perceptual validity. The expression transfer results from a human to two stylized characters are shown in Fig.5, which shows the generalizability of our algorithm in generating the same expressions on different characters having annotated training data. We tested the expression recognition on input human expressions and transferred expressions using 30 MT test subjects. The plot shown in Fig. 5 (d) shows high correlation between MT agreement for recognized expressions on ‘Mery’ (Fig. 5 (b)), which confirms the accurate perception of the intended expression transfer. We obtained a very similar plot for ‘Ray’ (Fig. 5 (c)).



(a) Input human images

(b) Output ‘Aia’ expression results

Figure 7: Consistent human expression transfer to primary character. This example shows (a) six different human images with the joy expression (b) transfered to the primary character ‘Aia’.

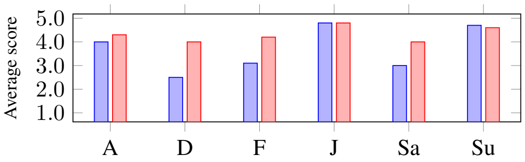


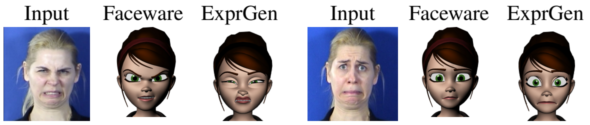
Figure 8: Quantitative comparison of expression transfer results of Faceware (blue bars) and ExprGen (red bars).

5.2.2 Multiple Humans to a Single Character

We generated the same expression class on a single primary character from different human inputs as illustrated in Fig. 7, showing that our algorithm is consistent in transferring the expressions even when there is variation in the human input examples.

5.2.3 Comparison with Faceware

ExprGen generates expressions with greater perceptual validity than popular commercially available software packages. We compared ExprGen with the award-winning Faceware technology [18], because it is the only feasible and comparable system that has the same input and output modality as ExprGen. Faceware includes *Analyzer* to convert human facial performance from a sequence of input images into motion capture data and *Retargeter* to map the captured data to the blendshapes of the 3D character face rig by manually creating an expression set for the character. Fig. 8 shows the comparison of average scores obtained for different expression classes when 30 MT test subjects were asked to rate the closeness of the expression generated on the character to the input human expression on a scale of 1-5, with 5 being the closest match. The average score over all classes for ExprGen is 4.31 versus an average score of 3.68 for Faceware. Fig. 9 shows the expression



(a) Input expression: disgust (b) Input expression: fear

Figure 9: Qualitative comparison of expression transfer results of Faceware and ExprGen (left to right: input human expression, Faceware output and ExprGen output).

transfer results of Faceware and ExprGen for two cardinal expressions. These results show that blendshape-mapping-based approaches often produce incorrect expressions (see Fig. 9(a)) or ambiguous expressions (see Fig. 9(b)) owing to the limitations of correspondence mapping. We did not compare with the results of Faceshift Studio [17], since it requires a depth camera to capture human facial motion and uses a different approach compared to our 2D human image to 3D stylized character rig mapping.

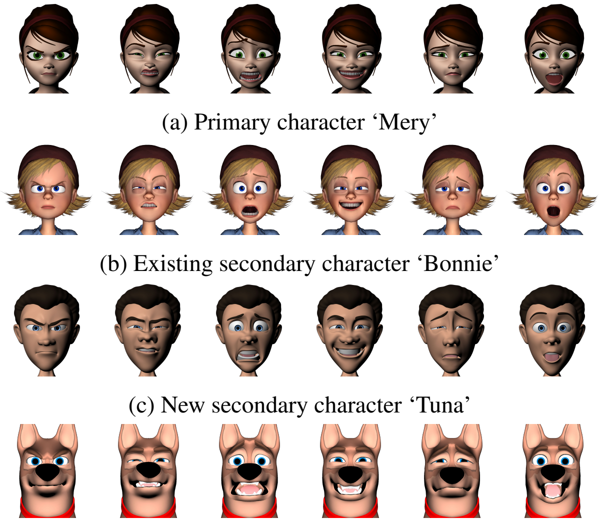
5.3. Character to Character Expression Transfer

In order to evaluate the performance of our character-to-character expression model, we selected 'Mery' as the primary character, 'Bonnie' as the existing secondary character (present in FERG-DB) and 'Tuna' and 'Cody' (non-human) as the new secondary characters (not present in FERG-DB). Fig. 10 (a) shows six randomly chosen cardinal expressions on the primary character used as test cases, and Fig. 10 (b), (c) and (d) show the facial expressions generated on the secondary characters at the output of the C-MLP. The results show that our network accurately learns the relationship between the 3D parameters of the characters, while maintaining the clarity of the expressions. Our network produces surprisingly good results for non-human characters as well, though the C-MLP is trained on only the key poses. However, the training examples for new secondary characters are critical to this approach, and there are two issues in selecting accurate training examples. First, when the new secondary character expression is perceptually valid but a similar expression does not exist for the primary character in the database (see Fig. 11 (a)), our method retrieves the closest possible match which may be inaccurate. Second, when the new secondary character expression is perceptually ambiguous (see Fig. 11 (b)), our method tries to find the closest match based on geometric features within the wrong expression classes, which may result in a wrong training example. Our future work will extend the training of new secondary characters by automating the process of generating large numbers of poses for each new character.

6. Conclusions and Future Work

We have demonstrated a novel multi-stage deep learning method to transfer human facial expressions to multiple 3D stylized characters that optimizes over expression clarity rather than over geometric markers. The resulting expressions, when validated by Mechanical Turk studies, show that our expression transfer clearly reproduces the input human expressions and generalizes to multiple human source expressions and multiple character targets. ExprGen has several practical applications including visual storytelling, video games, social VR experience and human-robot interactions. Our work provides the foundation for several future explorations, including learning expression intensity, adding animation etc. It will be interesting to add the concept of a universal primary character rig that is sufficiently powerful to create a full range of expressions and can be quickly extended to unusual character designs such as one-eyed or fantasy characters.

Acknowledgements. We thank Ezgi Mercan, Sachin Mehta and anonymous reviewers for their valuable comments and suggestions.

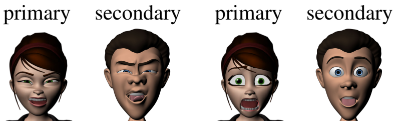

(a) Primary character 'Mery'

(b) Existing secondary character 'Bonnie'

(c) New secondary character 'Tuna'

(d) New non-human secondary character 'Cody'

Figure 10: Primary to Secondary character expression transfer results (left to right: anger, disgust, fear, joy, sadness and surprise). (a) 'Mery's' expression classes, Expressions transferred to (b) 'Bonnie', (c) 'Tuna', and (d) 'Cody'.



(a) (b)

Figure 11: Error cases in obtaining training examples for new secondary characters. (a) Matching is perceptually valid (both expressions are disgust) but geometrically incorrect, (b) Matching is perceptually invalid (expression on left is fear and on right is surprise) but geometrically correct.

References

[1] Character rigs for download. http://www.cgmeetup.net/forums/files/.

[2] Mass effect. https://www.masseffect.com.

[3] D. Aneja, A. Colburn, G. Faigin, L. Shapiro, and B. Mones. Modeling stylized character expressions via deep learning. In *Asian Conference on Computer Vision*, pages 136–153. Springer, 2016.

[4] S. Bouaziz, Y. Wang, and M. Pauly. Online modeling for realtime facial animation. *ACM Transactions on Graphics (TOG)*, 32(4):40, 2013.

[5] I. Buck, A. Finkelstein, C. Jacobs, A. Klein, D. H. Salesin, J. Seims, R. Szeliski, and K. Toyama. Performance-driven hand-drawn animation. In *Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 101–108. ACM, 2000.

[6] P. Burkert, F. Trier, M. Z. Afzal, A. Dengel, and M. Liwicki. Dexpression: Deep convolutional neural network for expression recognition. *arXiv preprint arXiv:1509.05371*, 2015.

[7] J.-x. Chai, J. Xiao, and J. Hodgins. Vision-based control of 3d facial animation. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 193–206. Eurographics Association, 2003.

[8] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 539–546. IEEE, 2005.

[9] E. Chuang and C. Bregler. Performance driven facial animation using blendshape interpolation. *Computer Science Technical Report, Stanford University*, 2(2):3, 2002.

[10] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, number EPFL-CONF-192376, 2011.

[11] T. F. Cootes, M. C. Ionita, C. Lindner, and P. Sauer. Robust and accurate shape model fitting using random forest regression voting. In *European Conference on Computer Vision*, pages 278–291. Springer, 2012.

[12] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 2106–2112. IEEE, 2011.

[13] H. Ding, S. K. Zhou, and R. Chellappa. Facenet2expnet: Regularizing a deep face recognition net for expression recognition. *arXiv preprint arXiv:1609.06591*, 2016.

[14] Dynamixyz. Performer suite. http://www.dynamixyz.com/.

[15] P. Ekman. An argument for basic emotions. *Cognition & emotion*, 6(3-4):169–200, 1992.

[16] I. Essa, S. Basu, T. Darrell, and A. Pentland. Modeling, tracking and interactive animation of faces and heads//using input from video. In *Computer Animation'96. Proceedings*, pages 68–79. IEEE, 1996.

[17] Faceshift. Faceshift. http://faceshift.com/studio/2015.2/.

[18] Faceware. Faceware live. http://facewaretech.com/.

[19] J. Huang and C. Pelachaud. Expressive body animation pipeline for virtual agent. In *International Conference on Intelligent Virtual Agents*, pages 355–362. Springer, 2012.

[20] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[21] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim. Joint fine-tuning in deep neural networks for facial expression recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2983–2991, 2015.

[22] S. E. Kahou, X. Bouthillier, P. Lamblin, C. Gulcehre, V. Michalski, K. Konda, S. Jean, P. Froumenty, Y. Dauphin, N. Boulanger-Lewandowski, et al. Emonets: Multimodal deep learning approaches for emotion recognition in video. *Journal on Multimodal User Interfaces*, 10(2):99–111, 2016.

[23] P. Liu, S. Han, Z. Meng, and Y. Tong. Facial expression recognition via a boosted deep belief network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1805–1812, 2014.

[24] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010.

[25] D. Lundqvist, A. Flykt, and A. Öhman. The karolinska directed emotional faces-kdef. cd-rom from department of clinical neuroscience, psychology section, karolinska institutet, stockholm, sweden. Technical report, ISBN 91-630-7164-9, 1998.

[26] I. Matthews, N. Kholgade, and Y. Sheikh. Content retargeting using facial layers, Jan. 26 2016. US Patent 9,245,176.

[27] I. Matthews, J. Xiao, and S. Baker. 2d vs. 3d deformable face models: Representational power, construction, and real-time fitting. *International journal of computer vision*, 75(1):93–113, 2007.

[28] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn. Disfa: A spontaneous facial action intensity database. *Affective Computing, IEEE Transactions on*, 4(2):151–160, 2013.

[29] Mixamo. Face plus. https://www.mixamo.com/faceplus/.

[30] A. Mollahosseini, D. Chan, and M. H. Mahoor. Going deeper in facial expression recognition using deep neural networks. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE, 2016.

[31] J.-y. Noh and U. Neumann. Expression cloning. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 277–288. ACM, 2001.

[32] OptiTrack. Expression. http://optitrack.com/products/expression/.

[33] M. Pantic, M. Valstar, R. Rademaker, and L. Maat. Web-based database for facial expression analysis. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 5–pp. IEEE, 2005.

[34] F. E. Pollick. In search of the uncanny valley. In *User Centric Media*, pages 69–78, Berlin, Heidelberg, 2010. Springer.

[35] J. M. Saragih, S. Lucey, and J. F. Cohn. Real-time avatar animation from a single image. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 117–124. IEEE, 2011.

[36] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[37] Y. Seol, J. Seo, P. H. Kim, J. Lewis, and J. Noh. Artist friendly facial animation retargeting. In *ACM Transactions on Graphics (TOG)*, volume 30, page 162. ACM, 2011.

[38] X. Tan and B. Triggs. Fusing gabor and lbp feature sets for kernel-based face recognition. In *International Workshop on Analysis and Modeling of Faces and Gestures*, pages 235–249. Springer, 2007.

[39] P. Tassi. ”mass effect: Andromeda” review (ps4): Every man’s sky, March 2017.

[40] X. Xiong and F. Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 532–539, 2013.

[41] U. U. Yetiskin. Tuna rig. `https://www.behance.net/gallery/31141085/Tuna-Rig-for-FREE`.

[42] Z.-L. Ying, Z.-W. Wang, and M.-W. Huang. Facial expression recognition based on fusion of sparse representation. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, pages 457–464. Springer, 2010.

[43] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[44] X. Yu, J. Yang, L. Luo, W. Li, J. Brandt, and D. Metaxas. Customized expression recognition for performance-driven cutout character animation. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–9. IEEE, 2016.

[45] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4353–4361, 2015.

[46] T. H. Zavaschi, A. S. Britto, L. E. Oliveira, and A. L. Koerich. Fusion of feature sets and classifiers for facial expression recognition. *Expert Systems with Applications*, 40(2):646–655, 2013.

[47] R. Zemeckis. The polar express. 2005.