FamilyGAN: Generating Kin Face Images using Generative Adversarial Networks

Raunak Sinha[1], Mayank Vatsa[2], and Richa Singh[2]

IIIT-Delhi, India, IIT Jodhpur, India
raunak15075@iiitd.ac.in, mvatsa@iitj.ac.in, richa@iitj.ac.in

Abstract. Automatic kinship verification using face images involves analyzing features and computing similarities between two input images to establish kin-relationship. It has gained significant interest from the research community and several approaches including deep learning architectures are proposed. One of the law enforcement applications of kinship analysis involves predicting the kin image given an input image. In other words, the question posed here is: "given an input image, can we generate a kin-image?" This paper attempts to generate kin-images using Generative Adversarial Learning for multiple kin-relations. The proposed FamilyGAN model incorporates three information, kin-gender, kinship loss, and reconstruction loss, in a GAN model to generate kin images. FamilyGAN is the first model capable of generating kin-images for multiple relations such as parent-child and siblings from a single model. On the WVU Kinship Video database, the proposed model shows very promising results for generating kin images. Experimental results show 71.34% kinship verification accuracy using the images generated via FamilyGAN.

Keywords: Kinship, image generation, generative adversarial networks, deep learning

1 Introduction

The prevalent discourse on kinship facial-analysis is determining if two individuals are related (kins) through given face images. This analysis extends to predict the possible relation between given individuals such as father-daughter, mother-son, and mother-daughter. Such relations are ascertained through leveraging and understanding common facial features [6], [7], [15]. In this research, we are exploring the scantly addressed question related to kinship analysis and predicting looks of possible kin of an individual (Fig. 1).

For cases of missing persons and long-lost relatives where kin-images were not available to compare, a possible kin-image can potentially assist in speeding

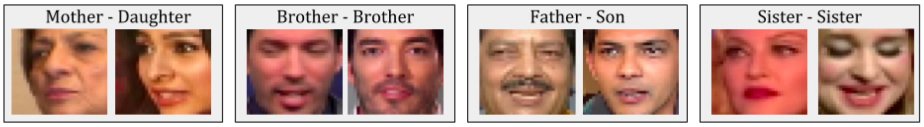

Fig. 1. Relation samples from WVU Kinship Video Database. FamilyGAN is learning to generate kin face images by understanding the facial heredity hierarchy in such relations and applying appropriate transforms.

Fig. 2. Sisters separated at birth[3].

Fig. 3. Suspects for the Boston Bombing[4]. The two individuals were later identified to be brothers.

the search[1],[2]. For a recent case of long-lost sisters at Greenwood, USA[3] (Fig. 2), where sisters were separated at birth,"probable kin-image" could assist this search. Kinship verification can help investigations such as Boston Bombing[4] (Fig. 3). With initial images of suspects, kinship verification could have helped them to ascertain relations and conduct targeted search for suspects, but what if the images of one of the brothers was missing. Kinship image generation can help synthesize possible family members.

We are keen on understanding the hierarchy of facial features amongst relations. Fabricating possible face image of kin given only the face image of a person requires capturing and reproducing dominant transforms observed in different relations. Applying an appropriate transform for different individuals is essential in the creation of images that can possibly be the face images of kins. Kin feature heredity varies extremely with a single family e.g. feature heredity between a mother-daughter is different than mother-son. This feature heredity also varies amongst different families as genetic matter shared between two pairs of mother-daughter is highly conditional. Such large variations make it hard to observe global patterns for fabricating kin images. While generating possible kin

[1] https://abcnews.go.com/Lifestyle/long-lost-brothers-discover-college-disbelief/story?id=51918769
[2] https://www.mirror.co.uk/3am/celebrity-news/rochelle-humes-reunites-long-lost-14977068
[3] https://abcnews.go.com/GMA/Family/adopted-woman-searches-long-lost-sister-learn-shes/story?id=56230030
[4] https://en.wikipedia.org/wiki/Boston_Marathon_bombing

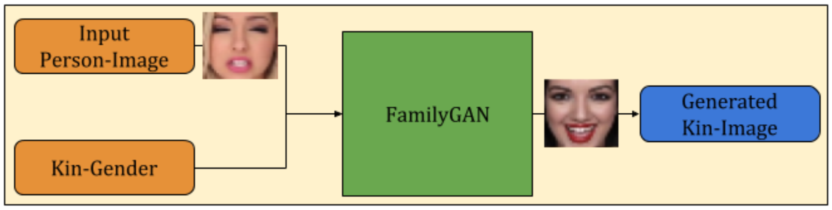

Fig. 4. Conceptualizing kin image generation task. FamilyGAN takes image of a person and kin-gender as input to generate possible kin image as output.

images extreme emphasis is required on physical features and type of relation for suitable transforms.

We formulate FamilyGAN to address the complex problem of kinship image generation. FamilyGAN is successful in learning intricate feature heredity and administering apposite feature transforms to generate the possible image of kin with just the input image of a person and the relation to be generated (Fig. 1). FamilyGAN is simultaneously trained to identify and verify kin relations between the given individuals as a by-product of learning to generate kin images. The research contributions are as followed:

- The proposed FamilyGAN model learns and understands kin feature heredity. The model is capable of administering learned feature transforms on the image of an individual to generate a possible face image of kin. The kin image is generated under the conditioning of kin gender (Fig. 4).
- We propose a novel optimization and loss for learning how to generate kin-images.
- We perform both qualitative and quantitative evaluation of kin face images generated by FamilyGAN on the WVU Kinship Video Dataset [14]. For quantitative evaluation we use two approaches: (i) evaluation using state-of-the-art kinship verification algorithms and (ii) face recognition algorithms.

2 Related Work

The problem statement of kin-image generation observes its foundation from the problem of kinship verification [19] and image generation. Kinship verification is determining if two individuals are related based on evidence of common physical features. Given the images of two individuals, kinship verification leverages the facial features of two individuals to answer the question - are the two individuals related (kins) or not. This binary classification can be further extended to multi-class classification, predicting the kin relation between given individuals such as father-daughter, mother-son, and mother-daughter [6], [7], [15]. The human face is formed by key features and regions such as eyes, nose, lips, cheeks, and face-shape, these facial-features are contingent on the genetic makeup of an individual [4], [5]. Therefore, to understand the kinship hierarchy we can discern the hierarchy in facial features amongst kin and use the learned hierarchy to generate possible kin images of an individual.

While the problem of kin image generation is derived from kinship verification, to the best of our knowledge, there are only two papers that attempt to generate possible kin images of a given individual. Ozkan *et al.* [17] use a cycle consistent GAN (CycleGAN) framework [22] to generate images of children by analysing images of parents. Their work is limited to generating images of children and do not model other kin-relations. Similarly Ghatas *et al.* [8] takes both parents (father and mother) as input and pass the concatenated information through a kin-feature predictor network. The predicted features act as input to a PGGAN [11] network for generating images of children for a given age.

3 Proposed FamilyGAN Model

GANs are generative networks that rely on adversarial training for learning an underlying distribution and generating new members of the learned distribution. These models can transform noise or alter input data to generate realistic-looking samples [9], [18], [16]. Various GAN architectures exist to model different kinds of distributions and problem statements, such as Deep Convolutions GANs (DCGANs) [18], WGAN [1], Conditional GANs (CGAN) [16], CycleGAN [22], Pix2Pix [10], and StarGAN [2].

The proposed FamilyGAN captures key kin-feature hierarchies using the proposed loss function. Three key components are driving the learning for achieving the desired transformations. The new formulated loss function (Equation 3, Equation 4) learns adversarial sample generation and kinship verification in cohesion. Learning features through kinship verification improves the training of the generator for this specific task. Furthermore, conditioning the generator on kin-gender and additionally conditioning the discriminator on kinship verification samples is not present in the current literature for a kin-image generation. Finally, constricting the generator with an additional reconstruction loss helps the generator learn better transforms as this drives the generator to maintain the facial-integrity of generated samples. This ensures that the generated images resemble naturally occurring human faces.

3.1 FamilyGAN Model

FamilyGAN learns to perform facial feature transformation observing underlying kin hierarchy. The transformations learned are kin-gender specific, where the model explicitly learns appropriate kin transformations for female kin relations and male kin relations. This is achieved by conditioning the generator on kin-gender. Kin feature hierarchy is dependent on kin relation and gender relations. Kins with same-gender relations (such as father-son, mother-daughter) have a higher correlation of physical features [21]. Thus, we are focusing on generating images of the same kin-gender so that FamilyGAN learns strong discriminating features for each kin-gender.

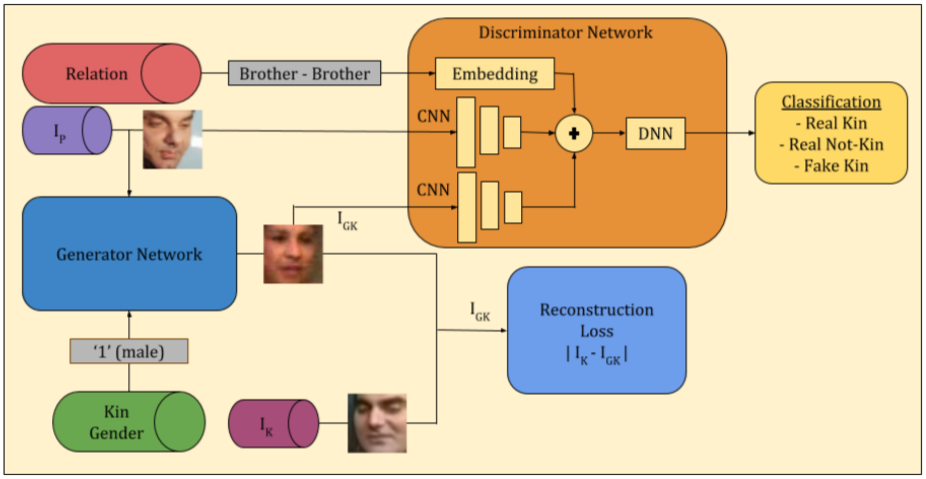

Fig. 5. Architecture of FamilyGAN model. Generator takes image person I_P and Kin-Gender as input and generates kin-image I_{GK}. Reconstruction loss between I_{GK} and I_K (real-kin) guides training. The discriminator takes I_P, I_{GK} and kin-relation as input to calculate an adversarial loss and also for kinship verification. Both these losses steer model training.

3.2 Generator

The generator of FamilyGAN is tasked with fooling the discriminator by generating realistic face images of individuals that can be their kin relative. Not only is the generator producing images that should look like samples from the given data space, but it is also additionally focused on generating faces that follow a particular kinship hierarchy for facial features. The generator needs to learn both these aspects to fool the FamilyGAN discriminator in believing that the generated kin-image is the actual kin of an individual. One of the prime novelties of FamilyGAN is to incorporate the notion of facial-feature hierarchy in a GAN framework.

Kin-gender Input to the generator is the images of an individual I_P and kin-gender label vector R (0 for female kin and 1 for male kin). A convolutional neural network with residual connections is used to extract latent features from I_P. We use residual connections to counter the degradation problem because of network depth. These connections ensure the flow of information to deeper layers, without any non-linear activation on residual connections the information flows freely during both forward and backward pass. The learned features are combined with the kin-gender label vector (one-hot encoding). This combined vector is processed by deconvolution layers to generate an image of appropriate dimensions. This is how we generate kin-images I_{GK} conditioned on kin-gender.

Reconstruction Loss Another important component of our generator is the reconstruction loss between the generated kin-image I_{GK} and the actual kin image I_K. We use mean squared error to find this loss. By adding reconstruction loss to the adversarial loss of generator, FamilyGAN governs the generation of the kin-image I_{GK} to be closer to the actual kin image I_K. This constraint drives the generator to learn the underlying kin feature hierarchy for each kin-pair, generating more probable looking relatives. Along with learning kinship hierarchy, the kinship loss forces the generator to maintain the facial integrity of generated images. We observed that it is difficult to preserve this with other models such as DCGANs [18], WGAN [1], CGAN [16], CycleGAN [22], Pix2Pix [10], and StarGAN [2].

$$L_G = E_{K \sim p-K}, E_{z \sim p-P}[log(1 - D(G(z|K)|I_P, I_{GK/K}, R))] \tag{1}$$

$$L_{G'} = L_G + \lambda ||I_K - I_{GK}||_2^2 \tag{2}$$

Equation 1 captures the real vs fake loss for the generated kin image, this is similar to the standard optimization for generator networks. Kin-image generation is conditioned over kin-gender K. The generator is optimized over Equation 2. Reconstruction loss is added to the loss of the generator as an auxiliary loss.

3.3 Discriminator

We construct a 3-class classification objective for our discriminator function. Given a pair of kin-images with their corresponding kin-relations, the discriminator determines the appropriate class, classes being: $\{[real + true\ kin], [real + false\ kin], [fakegenerated + true\ kin]\}$. FamilyGAN is optimized for only a 3-class classification as the $\{[fake - false\ kin]\}$ class does not fit into our objective of generating realistic samples that follow pertinent kin feature hierarchy. The proposed loss captures notions of both kinship-verification and adversarial training. Kinship-verification is learned through a tradeoff between $\{[real - true\ kin]\}$ and $\{[real - false\ kin]\}$. Concomitantly, the discriminator learns to distinguish real images from fake by training on both real images of kin I_K (from database) and fake kin images I_{GK} (from the generator) in an adversarial setting. Deep CNN extracts latent feature from given image input, CNNs can extract the information while keeping the spatial information of image intact.

During fakeness detection, the discriminator is concerned only with predicting if the input facial image is real or fabricated. At this stage, the discriminator is concerned with determining the closeness of generated samples and the actual data along with validating kinship-relations through kinship verification. While training the discriminator for learning kinship feature hierarchy both actual-positive-pairs (image-of-person I_P, actual-image-of-kin I_K) and generated-positive-pairs (image-of-person I_P, generated-image-of-kin I_{GK}) along with negative-actual-pairs are processed. The discriminator also takes kin-relation (father-son, daughter-mother, sister-sister, and brother-brother) as input. The discriminator processes the image-pair along with the kin-relation to determine if the image-pair are valid kins or not (the pair has to be related by the given kin-relation).

As training progresses the discriminator becomes smarter at detecting minute details between real and fake images making the discriminator more powerful. Now, as the discriminator becomes more powerful it guides the generator better and in turn, trains the generator for more realistic looking images. In addition to the real-fake discrimination, incorporating kinship verification in the optimization function of the discriminator makes the discriminator learn kinship feature hierarchy while learning to detect fakeness. We propose a new discriminator loss function in Equation 3. The discriminator classifies input pair as x, the decision is conditioned on image of person I_P, generated kin-image I_{GK} / real kin-image I_K and kin-relation R. To learn the decision boundary Cross-Entropy loss is calculated for the predictions. c is the number of classes (3), $y_{x,c}$ is 1 if x equals c otherwise 0.

$$L_D = -\Sigma_{c=1}^{3} y_{x,c} log(\mathbb{P}(D(x|I_{GK}, I_P, R))) \tag{3}$$

Combined loss equation for GAN model is:

$$min_G max_D (L_{G'} - L_D) \tag{4}$$

3.4 Model Training

The generator and discriminator of FamilyGAN are trained in tandem. The generator is dependent on the discriminator's ability to understand how far generated samples are from real kin. In turn, the discriminator becomes more capable in distinguishing minute difference as the generator becomes powerful. The FamilyGAN discriminator, in a combined fashion, finds out fake images as well as performing kinship verification for a given pair of images and their gender-relation.

The discriminator has two separate training steps. To optimally learn kinship-features through verification, the discriminator is initially trained over a data-set of both positive and negative kin-pairs. During this phase boundary between the [real]+[true kin] and [real]+[false kin] are learned. This lets FamilyGAN focus on learning optimal facial kinship features for guiding generation. For the second phase, the discriminator is retrained on only positive kin-pairs from the real-data as well as generated kin-images. During this step, the discriminator is being trained contemporaneously with the generator. Through this step the discriminator learns to optimize decision boundary for [real]+[true kin] and [fake]+[true kin].

To learn the kinship feature hierarchy transforms, we experimented with using the actual kin relations classes (for example father-son, mother-daughter) as conditional input to the generator. Such generation would provide more nuanced control concerning the kin feature transform, but FamilyGAN was not able to converge with such conditioning. The facial feature hierarchy follows some ubiquitous patterns amongst the same gender but the feature hierarchy may not be similar for all pairs, e.g. different mother-daughter pairs observe different feature transforms dependent on their gene. Such conditioning on gender separates the learning space, as it does not have to learn more constricted feature transforms based on kin-relations that may not follow generic patterns. Nuances for feature transforms are dependent on the input face image of the individual. FamilyGAN relies mainly on three key components, which are:

– The novel loss function (Equation 4) learns to optimize FamilyGAN jointly over kinship verification and kinship generation. This provides more supervision.
– Conditioning the generator on kin-gender as the feature hierarchy is more readily observed amongst kin-gender relations as compared to specific kin relations (e.g. father-son, mother-daughter), so the model can capture kin-gender transforms better.
– Inspired from autoencoders, a reconstruction loss is used to maintain the facial features of generated images. This constricts the generation of kin that looks like the true kin.

3.5 Implementation Details

True (real) positive and negative samples are randomly shuffled in the training dataset. Fake positive kin samples are generated by conditioning the generator on the input image of the person and conditioning of kin-gender for each pair. Adam optimizer is used for both the discriminator and generator loss.

The generator of FamilyGAN combines input image I_K and the kin-relation by a simple addition operation and passed through a series of convolution and deconvolution blocks to generate kin-images. Each convolution (downsampling) block consists of a convolution layer, instance normalization, ReLU activation, and residual connection. Whereas, the deconvolution (upsampling) blocks consist of a transposed convolution layer, ReLU activation, and instance normalization. The output of the generator is passed to the discriminator and also used to determine reconstruction loss.

For the FamilyGAN discriminator, each convolutional block consists of a convolutional layer, LeakyReLu activation, and Dropout layer. Inputs to the discriminator are passed to four such convolutional blocks before propagating them through a deep neural network for classification. The three inputs to the discriminator are processed separately before combining them for further propagation (Fig. 5). Two separate CNN networks process image of input-person I_P and image of kin I_{GK}/I_K. An embedding layer is used to transform the relation vector. A simple concatenation of extracted latent-feature vectors is then passed forward for processing.

4 Experimental Analysis

To evaluate the performance of the proposed FamilyGAN approach, we have used WVU Kinship Video database [13]. This section first briefly presents the database and protocol followed by the results.

4.1 WVU Kinship Video Database

This dataset contains video clips of individuals and kin-relationship information for positive and negative kin pair. The pairs have been divided between testing and training. For each individual, image frames are extracted from the video footage. Positive kin-pairs have the correct kin relations, whereas negative kin-pairs have false kin relations mentioned.

- 141 positive, 141 negative kin sets (videos) for training
- 214 positive, 214 negative kin sets (videos) for testing

There are seven types of kin-pair relations in the dataset mother-daughter, mother-son, father-daughter, father-son, brother-brother, sister-sister, and brother-sister. The database has majorly same gender kin cases, i.e. mother-daughter (21.28%), father-son (21.28%), brother-brother (11.34%), and sister-sister (13.47%). For cross-gender cases, the total cases are around 32%, i.e. mother-son (7.80%), father-daughter (16.31%), and brother-sister (8.52%). The Several image frames for each individual are extracted and filtered, the total number of positive pairs possible is 33,965,699. We find limited data for cross gender relations, such as mother-son and brother-sister [14].

The dataset contains extreme pose variations and that makes the underlying data distribution highly complex to learn. For pruning pose variance, we use pose estimation. For each individual, we choose 50 facial images (after pose estimation) providing $2,500$ image pairs for each kin set. We construct positive and negative kin pairs through the same process, to ensure uniformity and avoid unwanted bias. Both sets of kin pair (negative and positive) are necessary for proper optimization of the loss function. We follow the same procedure for generating test kin image pairs from 214 positive and negative testing kin sets.

The final processed training dataset consists of $462,500$ kin pairs (both positive and negative kins). The dataset provides meta-data of the kin relation for each pair. Kin type is crucial for training the discriminator to predict kin class correctly (kinship verification). Introducing kin verification loss in the overall loss function of the discriminator allows it to be more partitioned in learning latent features described in Section 3.4. We create additional kin-gender labels that we input to the generator. The generation of kin images is conditioned on the kin-gender, where female genders such as mother-daughter or sister-sister are given a label 0 while male genders such as father-son or brother-brother are given a label 1.

4.2 Results

Evaluating models for kin-image generation is a challenge. Kin-image datasets are not comprehensive in multiple regards. When generating kin images in an unconstrained environment, the generated image may belong to a certain point of time to the actual-kin-pair image. Additionally, when generating kin-images, a person may have multiple possibilities of kins based on different feature transforms. For example, a person may have 3 sisters and the evaluation dataset may

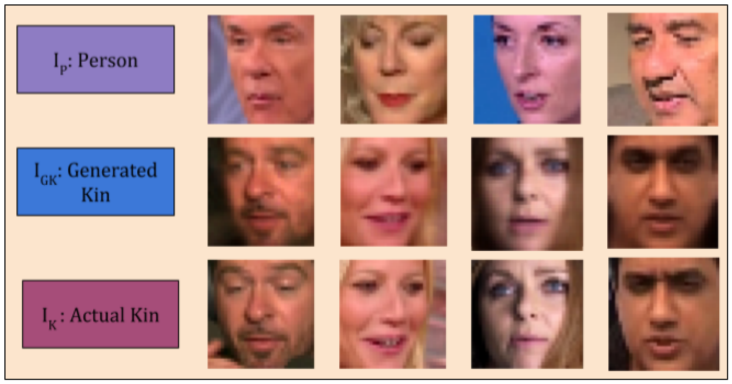

Fig. 6. Kin-samples generated from the training set, using the proposed FamilyGAN.

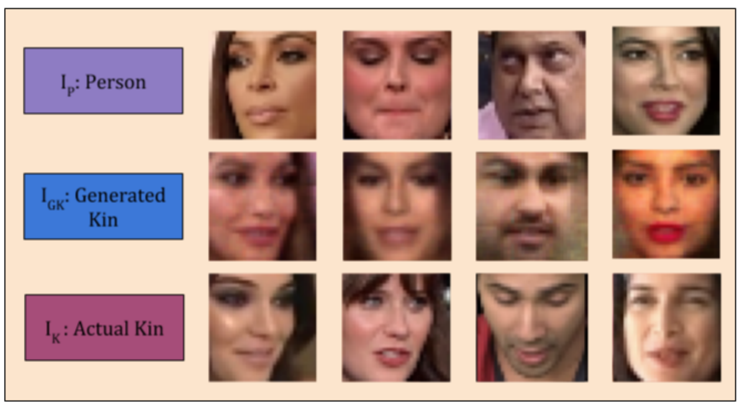

Fig. 7. Kin samples generated from the unseen (testing) set, using the proposed FamilyGAN.

contain samples of only 2. What happens if the kin-image generated resembles the third sister? To address these concern we propose two evaluation techniques. First, understanding the closeness in features of generated kin with input and real kin through a Siamese framework for kinship classification. Secondly, we evaluate generated kin-images through a state-of-the-art kinship verification model. Such a model is adept at understanding underlying notions of kinship relations in a given dataset and has fewer chances of giving a false positive. So we don't need to generate an image that is exactly similar to the real-kin pair for establishing that FamilyGAN is successfully generating kin-images. Furthermore, we provide with qualitative analysis of FamilyGAN and an ablative study it's various components.

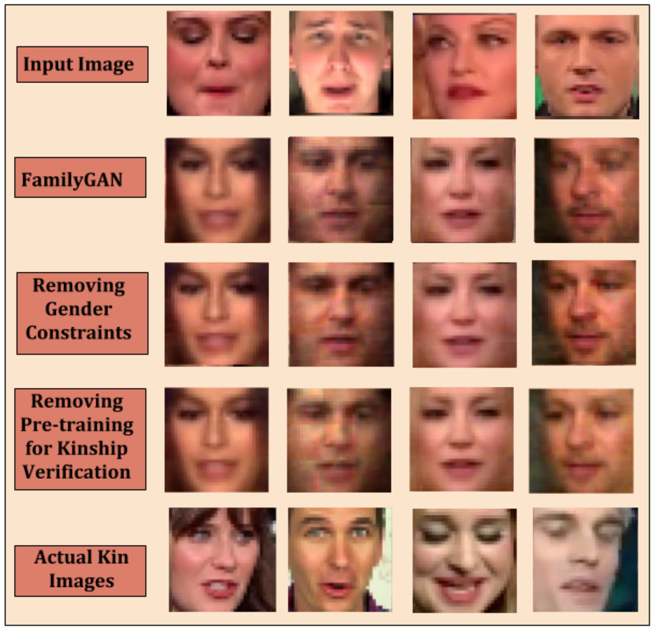

Fig. 8. Comparing effects of various FamilyGAN components for generating kin images on unseen (testing) data.

Qualitative analysis of kin image generation FamilyGAN successfully learns the facial feature hierarchy transforms. Transforms learned are used to generate possible kin images for a given individual. The generation is controlled through kin-gender as conditional input to the generator, this lets the generator use an appropriate transform for different kin-gender. The generation is dependent on the input image distribution, so each generated kin is dependent on the initial facial features of the input person. The dependence on the input image allows FamilyGAN to generate unique possible kin images, this dependence also enables FamilyGAN to generate realistic kin-image for previously unseen individuals (Fig. 7). Fig. 6 shows that the generator is learning the correct feature transforms during training and is able to model the observable kin feature hierarchy.

FamilyGAN drives the generator to produce such close results by including the reconstruction loss while training the model. From Fig. 6 and Fig. 7 we see that the network not only is able to learn the facial-feature transform but is also able to learn the pose variance in images. This again is achieved because of the addition of reconstruction loss.

A fascinating observation from Fig. 6 is that the generated kin-images have slight variations to the actual kin-images, which shows that the generator is not simply replicating actual kin images during training but learning sensible and generic kin feature transforms. Fig. 8 shows that FamilyGAN learns kin-

Table 1. LightCNN-Siamese network to quantitatively compare models for generation of kin-images. Reported values are the average-MSE between latent-feature representation of inputs to the siamese network for the testing data.

Model	Image Pairs			
	Real-Kin & Input (True Pairs)	Generated-Kin & Input	Real-Kin & Generated-Kin	Negative pairs (False Pairs)
FamilyGAN without Gender Constraints	2.775E-05	2.649E-05	2.503E-05	3.29E-05
FamilyGAN without Reconstruction Loss	2.564E-05	2.497E-05	2.460E-05	3.013E-05
FamilyGAN without pre-training for kinship verification	3.008E-05	2.954E-05	2.786E-05	3.722E-05
Proposed FamilyGAN	**2.478E-05**	**2.463E-05**	**2.334E-05**	**2.892E-05**

ship feature hierarchy in detail for each kin-gender. Specific feature transforms for different facial features are learned. As FamilyGAN learns kin-gender based feature transforms for each kin-pair, the transforms learned are generic to the kin-gender which makes the model useful to generate possible kin images even if the input person was not seen before. Not only are hierarchical relations in facial features learned, but are also learned for skin tone, hair type, hair color, eyebrow shape, and eye color.

To demonstrate the feasibility of FamilyGAN we show the result of kin generation for unseen samples from the testing data (Fig. 7). Generated kin images for unseen samples have facial features resembling the input person. The possible kin images show resemblance to the actual kin image but are closer to the input individuals. This happens because FamilyGAN is learning generic but specific for a kin-gender, making the transformations are not specific to any single kin pair. This allows FamilyGAN to generate more likely kin images for any unseen input. The resemblance between generated and actual kin shows that learned transforms capture the notion of kin hierarchy. In section 4.2 we determine experiment and discuss result for a quantitative evaluation of generated kin images.

Quantitative Analysis of LightCNN-Siamese Kin Image Distance To evaluate the generated images with rigor, we formulate an experiment that can quantify how similar are the generated kin-images I_{GK} to both input person I_P and real-kin I_K. The following quantitative experiment is performed to evaluate the similarity. We train a LightCNN [20] based siamese network [3], [12] to capture the closeness of true kin-pairs (I_P and I_K) in terms of facial features. To achieve this, we optimize the MSE-distance between the true image of the person and real-kin image from the training set. The distance between latent-feature representations from the two networks is a quantitative benchmark for kinship similarity. The distance between input person I_P and generated kin I_{GK} is now calculated to gauge the performance of the generator. We additionally find the distance between the generated kin I_{GK} and real-kin I_K to evaluate the closeness.

The distances for testing protocols are determined, the results are summarised in Table 1. We can see that proposed FamilyGAN outperforms other models and has the least distance between the input person and generated kin. We observe comparable distance between 'real-kin & input person (True)' and 'generated-kin & input person', which indicates that FamilyGAN is aware the appropriate feature transform that should be applied given the context. Additionally, the distance between 'real-kin & generated-kin' is the least amongst comparable pairs showing that the generated-kin is close to the real-kin in the embedding space.

This effectively shows us that FamilyGAN can learn and apply suitable kinship feature transforms such that the generated kin images are close to the true kin as well as input in terms of feature hierarchy. The power and utility of FamilyGAN can be observed by the images generated for unseen (testing) samples. Though the model has not seen the images before it can apply appropriate transform based on features and maintain kinship feature hierarchy.

Kinship Verification Performance Using the experimental protocol (frame-based) defined in Kohli et al. [14], we performed kinship verification experiments. We generated 3674 kin images of real subjects using FamilyGAN and computed the kinship verification accuracies (i.e. "positive pairs between real (input) - generated kin"). For these input images, we have real kin images that are used to compute verification accuracy of "real to real kinship positive pairs". Using Supervised Mixed Norm Autoencoder for kinship verification approach [14], we computed the positive pair accuracy and observed that for "real (input) to real kinship positive pairs", it is 74.06% whereas, for "real (input) to generated kin positive pairs", the accuracy is 71.34%. This experiment shows that the proposed FamilyGAN is able to generate images useful for automatic analysis as well.

5 Conclusion

Learning to generate kin face images by understanding the nuances of kinship facial feature as well as heredity patterns and when to apply appropriate transforms is an arduous task. FamilyGAN is a novel model that attempts to capture these complex feature hierarchy and govern the generation of possible kin face images. FamilyGAN conditions the generation of kin face images on kin-gender (Section 3.2) and the input face image of the person. The generative dexterity of FamilyGAN is analyzed qualitatively and quantitatively to show that the generated images are closely related to input face image of person and the real-kin face image in terms of facial features. FamilyGAN is adept at applying felicitous facial features transform to maintain kin feature hierarchy while observing relation (kin-gender) constraints. As a future work, we plan to extend the model to include kin-relation as well, to enable generating kin-images of different gender relations, such as father-daughter and mother-son.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. arXiv preprint arXiv:1701.07875 (2017)
2. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. arXiv preprint arXiv:1711.09020 (2017)
3. Chopra, S., Hadsell, R., LeCun, Y., et al.: Learning a similarity metric discriminatively, with application to face verification. In: CVPR (1). pp. 539–546 (2005)
4. Cole, J.B., Manyama, M., Larson, J.R., Liberton, D.K., Ferrara, T.M., Riccardi, S.L., Li, M., Mio, W., Klein, O.D., Santorico, S.A., et al.: Human facial shape and size heritability and genetic correlations. Genetics **205**(2), 967–978 (2017)
5. Crouch, D.J., Winney, B., Koppen, W.P., Christmas, W.J., Hutnik, K., Day, T., Meena, D., Boumertit, A., Hysi, P., Nessa, A., et al.: Genetics of the human face: Identification of large-effect single gene variants. Proceedings of the National Academy of Sciences **115**(4), E676–E685 (2018)
6. Dahan, E., Keller, Y.: Selfkin: Self adjusted deep model for kinship verification. arXiv preprint arXiv:1809.08493 (2018)
7. Fang, R., Tang, K.D., Snavely, N., Chen, T.: Towards computational models of kinship verification. In: Image Processing (ICIP), 2010 17th IEEE International Conference on. pp. 1577–1580. IEEE (2010)
8. Ghatas, F.S., Hemayed, E.E.: Gankin: generating kin faces using disentangled gan. SN Applied Sciences **2**(2), 1–10 (2020)
9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
10. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1125–1134 (2017)
11. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)
12. Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: ICML deep learning workshop. vol. 2 (2015)
13. Kohli, N., Vatsa, M., Singh, R., Noore, A., Majumdar, A.: Hierarchical representation learning for kinship verification. IEEE Transactions on Image Processing **26**(1), 289–302 (2016)
14. Kohli, N., Yadav, D., Vatsa, M., Singh, R., Noore, A.: Supervised mixed norm autoencoder for kinship verification in unconstrained videos. IEEE Transactions on Image Processing **28**(3), 1329–1341 (2018)
15. Lu, J., Hu, J., Zhou, X., Zhou, J., Castrillón-Santana, M., Lorenzo-Navarro, J., Kou, L., Shang, Y., Bottino, A., Vieira, T.F.: Kinship verification in the wild: The first kinship verification competition. In: Biometrics (IJCB), 2014 IEEE International Joint Conference on. pp. 1–6. IEEE (2014)
16. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
17. Ozkan, S., Ozkan, A.: Kinshipgan: Synthesizing of kinship faces from family photos by regularizing a deep face network. In: 2018 25th IEEE International Conference on Image Processing (ICIP). pp. 2142–2146. IEEE (2018)
18. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)

19. Wang, W., You, S., Karaoglu, S., Gevers, T.: Kinship identification through joint learning using kinship verification ensemble. arXiv preprint arXiv:2004.06382 (2020)
20. Wu, X., He, R., Sun, Z., Tan, T.: A light cnn for deep face representation with noisy labels. IEEE Transactions on Information Forensics and Security **13**(11), 2884–2896 (2018)
21. Xia, S., Shao, M., Luo, J., Fu, Y.: Understanding kin relationships in a photo. IEEE Transactions on Multimedia **14**(4), 1046–1056 (2012)
22. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)