CG-GAN: An Interactive Evolutionary GAN-based Approach for Facial Composite Generation

Nicola Zaltron
IT University of Copenhagen
Copenhagen, Denmark
nicolazaltra@gmail.com

Luisa Zurlo
IT University of Copenhagen
Copenhagen, Denmark
zurlo.luisa@gmail.com

Sebastian Risi
IT University of Copenhagen
Copenhagen, Denmark
sebr@itu.dk

Abstract

Facial composites are graphical representations of an eyewitness's memory of a face. Many digital systems are available for the creation of such composites but are either unable to reproduce features unless previously designed or do not allow holistic changes to the image. In this paper, we improve the efficiency of composite creation by removing the reliance on expert knowledge and letting the system learn to represent faces from examples. The novel approach, *Composite Generating GAN* (CG-GAN), applies generative and evolutionary computation to allow casual users to easily create facial composites. Specifically, CG-GAN utilizes the generator network of a pg-GAN to create high-resolution human faces. Users are provided with several functions to interactively breed and edit faces. CG-GAN offers a novel way of generating and handling static and animated photo-realistic facial composites, with the possibility of combining multiple representations of the same perpetrator, generated by different eyewitnesses.

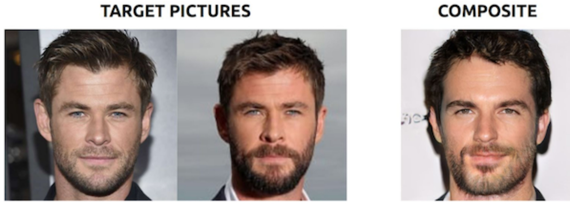

Figure 1: Example composite built using CG-GAN.

Introduction

Facial composites are portrait sketches of unknown individuals used in criminal investigation to identify a person. Two cognitive abilities are applied in the process: recall and recognition. Recall is the recollection of information for the composite construction. Recognition is the ability to recognize someone seen before, used during lineups (Zahradnkov, Duchoviov, and Schreiber 2016; Mancusi 2014). Compared to recall, recognition is an easier and stronger task, therefore most courts of law consider lineup identification as good evidence (Mancusi 2014).

Originally composites were drawn by forensic artists in consultation with victims and eyewitnesses, relying on detailed descriptions of their memories. More recent approaches involved mechanical and digital systems to improve the process and the resulting recognition rate.

There exist two major modern creation techniques. Simple systems merge single features (drawn or picked from a feature set) into a portrait. Others let the witnesses concentrate on the entire face, by selecting individuals as in a lineup (Zahradnikova, Sutova, and Schreiber 2017). This *holistic approach* is the result of psychological research suggesting that describing individual features causes lower identification rates (Zahradnikova, Sutova, and Schreiber 2017).

CG-GAN is designed to make the process artist-free to avoid communication issues and potentially lower retention time, and is empowered by the usage of recent machine learning techniques. In more detail, the approach is based on the *Latent Variable Evolution* (LVE) approach (Bontrager, Togelius, and Memon 2017), in which a Generative and Adversarial Network (GAN) is trained on a specific dataset and the space of images encoded by the GAN is then searched through an evolutionary computation approach. LVE has been applied to diverse domains such as the generation of fingerprints (Bontrager, Togelius, and Memon 2017), image generation (Bontrager and others 2018) and even the creation of Mario levels (Volz et al. 2018).

Bontrager and others (2018) showed that LVE can enable users to guide the search through a process known as interactive evolution (Takagi 2001). However, evolving towards a specific target image, which is a prerequisite for facial composite generation, has shown challenging (Bontrager and others 2018). This work goes beyond the previous state-of-the-art by extending LVE with the ability to freeze certain features discovered during the search, enabling a more controlled user-recreation of target images. The system exploits advantages of both constructive and holistic techniques. Figure 1 shows an example of a composite session that has been carried out with the usage of CG-GAN.

Related work

Facial Composite Generation

Police agencies are already using software for the creation of facial composites, such as Faces 4.0, which allows adding and editing features (IQ Biometrix 2003). However, this kind of software is not capable of reproducing facial features that have not been previously designed, is not holistic and requires the user to explore a vast feature database to construct a face. Other programs such as EvoFit (Frowd, Hancock, and Carson 2004) and EFIT-V (George et al. 2008) are based on an interactive evolutionary algorithm. These systems can produce high-quality composites but require complex mathematical functions to create the face. Every aspect that is not planned a priori can therefore not be produced.

Interactive evolutionary computation (IEC) is a particular form of AI-assisted creation, in which a human functions as the fitness function of an evolutionary algorithm. IEC is often used in subjective domains which are hard to define mathematically, such as in evolving two-dimensional images (Secretan et al. 2011), three-dimensional forms (Clune and Lipson 2011), musical compositions (Hoover et al. 2012), or agent behaviors (González de Prado Salas and Risi 2017). In an iterative fashion, the system presents a number of artifacts for the experimenter to evaluate; the user, in turn, responds by indicating which of these artifact(s) is preferred. The next generation of artifacts is produced through mutations and/or crossover to the underlying representation, from which the user selects their favorite artifact(s), again forming the basis for the next generation. While the conceptual idea of IEC is intriguing, there are several challenges hampering the practical usefulness of the technique. Often a large number of evaluations are needed to find the desired artifact, an issue known as user fatigue. Additionally, it can be difficult to find a particular artifact a user has in mind. Part of the problem can be traced back to the underlying representation, i.e. the employed genotype-to-phenotype mapping, which might thread the wrong balance between generality and domain specificity.

Generative models

Generative models are a branch of machine learning techniques, in which the objective is to generate content (LeCun, Bengio, and Hinton 2015). They include diverse image-related application areas such as completion, correction, production of variations, denoising, up-scaling, etc. In the work presented here, we employ Generative Adversarial Networks (GANs) (Goodfellow and others 2014), which are a class of ML algorithms that are trained in an unsupervised way. GANs make use of two neural networks (NN) to simultaneously train two models: a generator G estimates the data distribution while a discriminative model D estimates if samples are from the training set or synthetic. During training, the goal of G is to maximize the probability of D misclassifying samples, like in a minimax two-player game. Eventually, G recovers the data distribution and D becomes unable to distinguish generated content.

Many extensions have been proposed, such as Deep Convolutional Generative Adversarial Networks (DC-GANs) (Radford, Metz, and Chintala 2016), a class of Convolutional Neural Networks (CNNs); Auto-Encoder Generative Adversarial Networks (AE-GANs) (Makhzani et al. 2016); and Plug and Play Generative Networks (PPGNs) (Nguyen et al. 2016). This paper employs a recent extension to GANs called Progressive Growing of GANs (pg-GAN) (Karras and others 2017), which is a training methodology for GANs that allows generator and discriminator to grow progressively: Starting with low (4×4) spatial resolution, new layers are added as the training proceeds to model increasingly fine details (Karras and others 2017). This speeds up and stabilizes training, allowing the creation of high-quality images.

Figure 2: Latent variable evolution results from (Bontrager and others 2018).

In this paper, we employ a pg-GAN[1] pre-trained on the CelebFaces Attributes Dataset (Liu and others 2015), which contains 200,000 images of celebrity faces, annotated with 40 binary attributes. The dataset (or its HQ version; Karras and others, 2017) can be used for learning both the creation of facial images (Karras and others 2017; Karras, Laine, and Aila 2018; Guan 2018) and assigning attributes to face images (Guan 2018).

Latent Variable Evolution

The approach in this paper is based on *latent variable evolution* (LVE); the basic idea is shown in Figure 3. First a GAN is trained in an unsupervised way to generate specific target content. In the second phase, the space of content can be searched by applying evolutionary techniques to the latent vector fed into the GAN's generator. The first LVE approach was introduced by Bontrager, Togelius, and Memon (2017). In their work, the authors train a GAN on a set of real fingerprint images and then apply evolutionary search to find a latent vector that matches with as many subjects in the dataset as possible.

In another paper Bontrager and others (2018) present an interactive LVE system, in which users can evolve the latent vectors for a GAN trained on different classes of objects such as faces or shoes (an approach also employed by the popular Ganbreeder app (Simon 2018)). Because the GAN is trained on a specific target domain, it becomes a compact and robust genotype-to-phenotype mapping (i.e. most produced phenotypes do resemble valid domain arti-

[1]https://github.com/tkarras/progressive_growing_of_gans

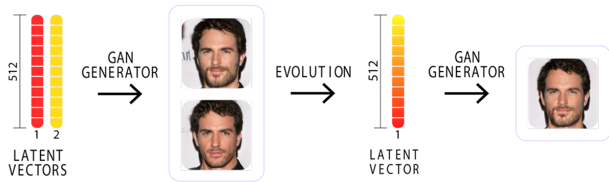

Figure 3: Latent Variable Evolution. Starting from a pre-trained GAN (left), the latent vector space (right) can be searched to create images with certain properties.

facts) and users were able to guide evolution towards images that sometimes resembled given target images. Such target-based evolution has been shown to be challenging with other generative representations (Woolley and Stanley 2011). However, the approach introduced by Bontrager and others (2018) does not allow to freeze discovered facial features, which limits the amount of control the user has during the search (Figure 2). For example, once a facial feature such as the beard looks just right and the user would only like to tweak the eyes, subsequent mutations to the latent vector will likely change both of these features.

Approach: CG-GAN

CG-GAN combines GANs with interactive evolution and additional features for fine-grained control (Figure 4). Through IEC users are able to explore solutions and control the input (i.e. the latent vector) of a pre-trained GAN's generator. Similarly to previous work by Bontrager and others (2018), this approach limits user fatigue by restricting the search space to the learned genotype-to-phenotype of the generator. While the approach by Bontrager and others (2018) implements a simple interface to visualize and select individuals, with a slider acting on the standard deviation of the mutation's noise, the approach introduced in this paper gives users fine-grained control. This advance enables the evolution of accurate composite sketches and is realized through an intuitive user interface and two essential approach additions: feature extraction and smart locks.

Feature extraction. One challenge with GANs is the control of the output: although the synthetic content can have remarkable quality, it depends on the random noise that is fed into the generator. To gain precise control of individual composite features, our approach includes a method called Transparent Latent Space GANs (TL-GAN) (Guan 2018), which consists in training a *classifier* CNN to correlate the changes in the input space to the changes in the output. Here facial features are measurable and representable characteristics of human faces. Facial features represent the major mean for communicating the traits recalled by eyewitnesses and for comparing and describing differences. Moreover, such features are essential for manually editing faces and can also enable a more controlled evolutionary process. Indeed, features can be interpreted as axes along which any face can be modified, both manually or automatically, to edit specific traits of a given face. The approach, based on TL-GAN, consists in discovering feature axes in a trained generator's latent space so that a vector can move along an axes to morph

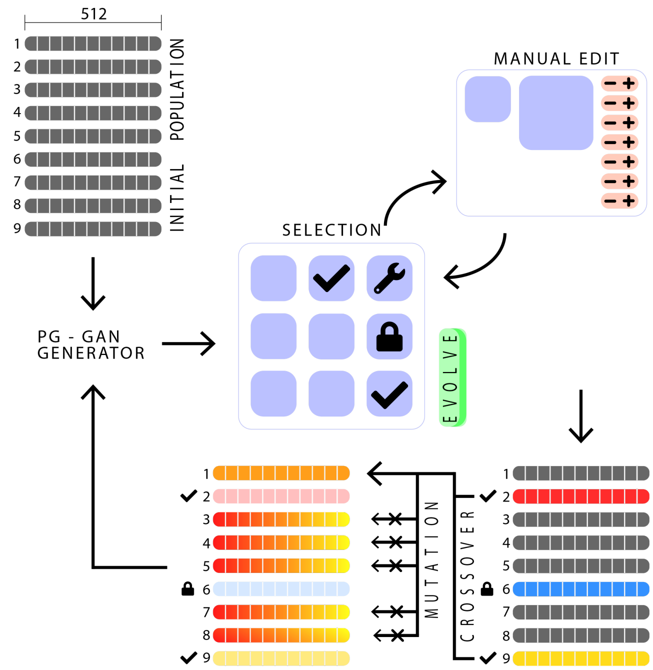

Figure 4: CG-GAN overview. The approach starts by presenting the user with a selection of varied images by initially inputting random latent vectors in a pg-GAN generator. Next the user can evolve faces interactively, manually edit, or lock features. The process is repeated for a number of generation, until the desired composite is created.

an image's feature. This is achieved by finding correlations between noise vectors and image features, through supervised learning (Guan 2018). For CG-GAN, we re-created our own models and feature axes.

Feature locks and smart locks. An issue with editing feature axes is that one change can potentially modify other correlated features (e.g. decreasing beard makes the face more female-like). As in TL-GAN, we disentangle features (i.e. axis in multidimensional space) by orthogonalizing axes, using their projections over different directions (Guan 2018). In other words, components that interfere with that axis are subtracted, which creates a new axis that is somewhat similar to the non-orthogonalized one. CG-GAN allows users to lock features: when a feature gets locked all the other ones are re-calculated by subtracting their projection over locked axes, thus avoiding their interference with any of the currently locked ones. CG-GAN also introduces *smart locks* to assist the user by locking sets of features based on their correlation. In more detail, the cosine similarity is computed for each pair of axes and when a feature is smart-locked both that feature and all the ones *strongly correlated* to it (here defined as exceeding a cosine similarity threshold of ± 0.5) are locked/unlocked. This addition allows users to modify traits without overwriting others that are likely to change. As an example, by *smart-locking* the gender attribute, also beard, moustache, hair length, makeup are locked, among others.

CG-GAN User Interface

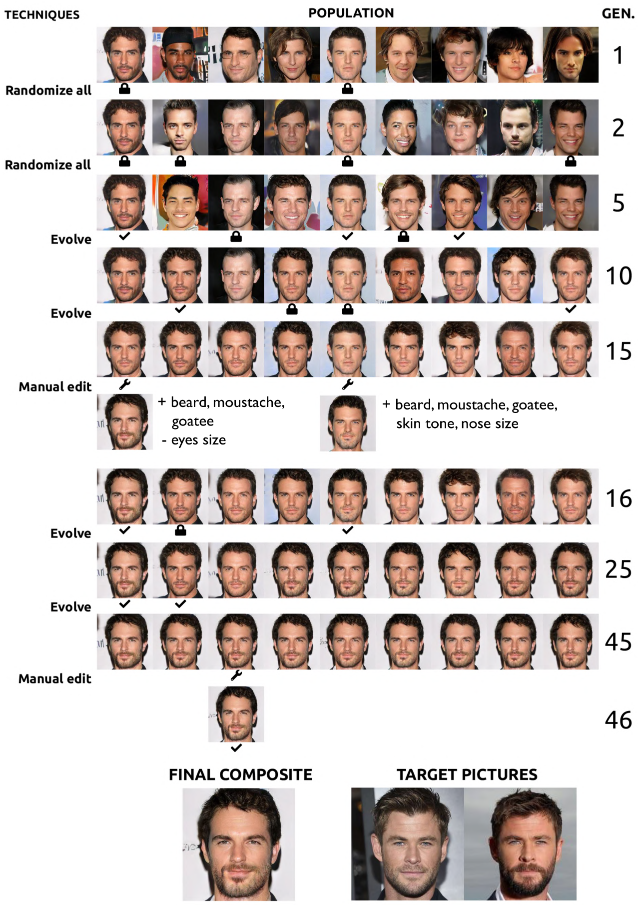

Figure 5: Shown is an example of a user session, including the employed user actions across generations. Given a target (Chris Hemsworth), CG-GAN was used to construct the composite in 46 generations, taking approximately 35 minutes. The user employed randomization, evolution and manual edits. Generations skipped between adjacent rows indicate the same functionality has been used multiple times.

A session starts with nine random faces that resemble gender and age chosen in a start-up panel (Figure 6). The user can *select* images to evolve them, *lock* them to preserve them, create new random faces or choose to manually *edit* them. Selected images are kept and used as parents for the next generation. Locked images are only kept unvaried. The first *free* image (not locked nor selected) is replaced by the crossover-generated offspring, which is the exact average of the selected faces' latent vectors. All the remaining *free* images are replaced with images created by mutating their underlying latent vectors.

Three mutation types are available: *random changes*, *one unlocked feature* and *every unlocked features*. *Random changes* adds Gaussian noise to the latent generator input, causing random changes in the faces proportionally to the chosen amount of noise that can be adjusted using a slider. The same applies to all other mutation types. The *random*

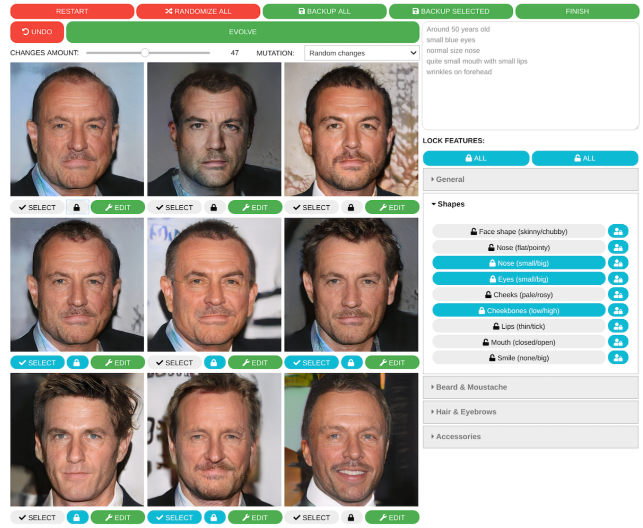

Figure 6: The main UI supports (1) randomization, (2) three kinds of mutation, (3) to lock and unlock features, and (4) to lock, select and manually edit faces. Next to the feature *locks* are also the *smart locks*. Features are changed according to the selected mutation type and a slider controller the amount of change.

changes mutation does not differentiate what aspects of the faces are changed. *One unlocked feature* and *every unlocked features* act on specific traits. The user can choose features that should or should not change. *One unlocked feature* changes exactly one feature by a certain amount. *Every unlocked feature* changes all the unlocked features by a lower amount, which is proportional to the desired *changes amount* and inversely proportional to the number of features that are being changed. The amount depends on a Gaussian distribution with μ and σ defined as:

$$\mu = \frac{20 \times \text{desired\_changes\_amount}}{F} \qquad \sigma = \frac{\mu}{3} \quad (1)$$

where F is set to 1 in case of *one unlocked feature* and $F = \min(\max(1, (0.8 \times \#\text{unlocked\_features})), 8)$, in case of *every unlocked feature*. The specific values were fine-tuned through prior experimentation.

As an example, let's assume all features are locked but three: beard, hair length, hair color. *One unlocked feature* will select one of them so that the output face will have either a different beard, hair length or color. *Every unlocked feature* instead will change all those by a lower amount, so that the output has both slightly different beards, hair length, and colors.

Advanced editing. At any time, the user can manually *edit* a face, by accessing the dedicated panel. The *changes amount* slider and feature locking system work analogously to the ones in the *Main panel*. The additional functionality is the possibility of acting over single feature axes, using a - and + button. A preview is updated at each modification and presets can be saved and loaded. The user can *save* the changes to overwrite the selected individual in the *Main panel*, which can be further edited or evolved.

Exporting results. When the user is satisfied with the produced results, the session can be concluded and results exported by selecting *finish* after having selected one or more images. In case of multiple selections, an animation that interpolates between them is created. Figure 7 shows a sprite-sheet portion of such an animated GIF. The user can navigate through the frames and export the whole animation or any frame. In the case of our user study we create a single image out of an animation, which is the average of all the selected latent vectors (Figure 8).

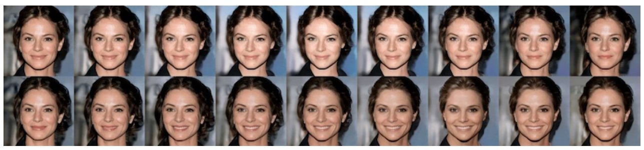

Figure 7: Sequence of frames of an animated composite.

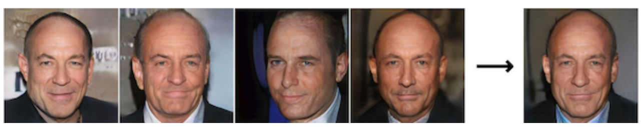

Figure 8: A merged session showing four composites (left) and their merged composite (right). The merge composite is generated by taking the average of all latent vectors.

Merging multiple witnesses' sessions. If more witnesses create composites, it can be useful to present a combined composite to the public (Davis et al. 2010). CG-GAN allows to load all data of a case with multiple witnesses, and the combined composite is computed based on a simple or weighted average of the underlying latent vectors. If weighted, composites can be assigned weights based on the type of witness[2] that created them. This feature was not extensively evaluated here but offers an exciting opportunity for future work in composite sketch generation.

Experiments

A preliminary testing phase was performed, which aimed at understanding how non-experts interact with the system. The study involved a diverse group of users and allowed us to fine-tune the interface; some results are shown in Figure 9.

For the final user test, participants were divided in two groups: 13 *constructors* created composites and 55 *evaluators* evaluated them[3]. Participants were recruited among friends, family and students in order to ensure variation in age, habits and skills.

[2]Witness types are defined as Active, Passive and Inactive according to The Police Composite Sketch (Mancusi 2014).

[3]Names *constructor* and *evaluator* are inherited from the gold-standard protocol for laboratory evaluation (Frowd et al. 2005).

Figure 9: Some results from a preliminary testing session. On top are the targets, on the bottom the built composites.

Regarding the construction of the composites, recall issues were avoided by letting users check the target over the whole duration of the test session. This choice is driven by our goal of evaluating the potential of a novel approach, rather than evaluating a realistic usage. Indeed a realistic usage scenario would involve factors that could bias the evaluation; it would be harder to pinpoint if a failure of the system is due to the CG-GAN not being capable to create a composite, or to the user having difficulty in remembering the face.

Target images for the user to recreate were generated using the GAN's generator: Latent vectors were randomized to export four faces not containing graphical imperfections (Figure 11). Strict rules were set on the timing: every user had 15 minutes to try the software before starting. The sessions lasted up to 30 minutes and data was collected every 10 minutes, to compare different phases of the process.

Testers filled in a survey after every test session. The questions aimed at determining which functions were used, the level of understanding of such functions and the users' satisfaction with their result. The questionnaire, its results, a demo and the code for the experiments in this paper can be found at: https://github.com/LuisaZurlo/CG-GAN.

Evaluation metrics

The objective evaluation of the system is a complex task as it depends on subjective recognition abilities (Zahradnkov, Duchoviov, and Schreiber 2016). To overcome this issue we use two separated evaluation metrics:

Similarity score: 27 *evaluators* assigned a similarity score from 0 to 100 to each composite. We emphasized that the score is meant to represent how likely the images are to depict the target person.

Recognition Rate: This measure tests the real purpose of the composites. Each tester (28 total) was given four sets of images. Each set composed of one composite and a lineup of five suspects (Figure 10). The users ranked the suspects based on their similarity to the composite. Since the context needs accurate predictions, we only considered the first choice, using the ranking to calculate a recognition rate r:

$$r = \frac{\#\,\text{Rank 1}}{\#\,\text{Votes}} \times 100 \quad (2)$$

It is important to note the potential bias of the metric, as it depends on the shown lineup and the user's subjective perception. To mitigate this bias, the target images (Figure 10) were generated to look *reasonably different* by taking the original GAN-generated image and producing four variations of it by adding a fixed amount of Gaussian noise to the original latent vector. These reasonably different faces share the majority of common traits, but are distinguishable from each other. We discarded images that only differ based on light, exposure, pose, or a too limited set of traits.

Figure 10: Different faces shown to users in the *recognition* experiment. Each row represents a line-up, composed of a target and four variations.

Results

Figure 11 shows the 16 composites generated by the different users for the four selected target samples. Scores are shown in Figure 12. Overall the results were promising, with 10 out of 16 composites recognized by at least 50% of all users, and 7 out of 16 composites recognized by at least 75% of all users. The average *similarity* score was 47.51%.

Comparing the results directly to the previous baseline by Bontrager and others (2018) is difficult since they did not include any recognition test in their experiments. However, users in their study reported a relatively low score in their ability to reproduce a given target face (an average score of 2.2 out of 5). Also, because the users could not freeze any features, they had to find less efficient workaround strategies that might have lead to some frustration. Visually, the quality of the resulting faces is also less convincing (Figure 2), while the fine-grained control of CG-GAN allowed users to create face composites matching the target image to a higher degree. Overall, users rated their experience with our system with a score of 7.31 (out of 10) and reported a score of 7.18 when asked how confident they were with the different available functions in the system. Users rated the accuracy of the resulting composite with 6.81.

Overall we noticed that after some generations users started to find traits that, according to their perception, resembled the target. Interestingly, the four targets have noticeably different scores. Images resembling target 1 are in general much more similar to the target compared to the ones representing targets 3 and 4 (Figure 11). These results were mirrored by the results from the questionnaire, in which users noted that some targets are easier to recreate than others. Very high variance on votes for the same composites

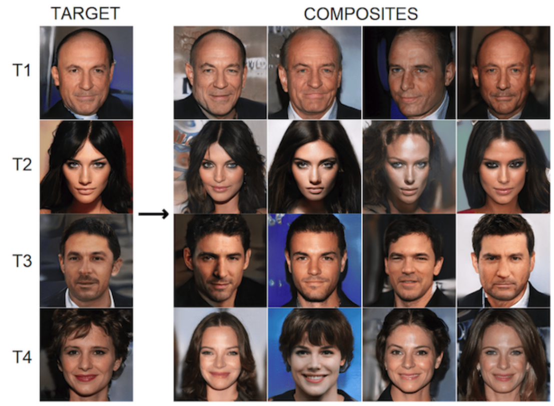

Figure 11: Target samples (left) and the relative composites built by constructor testers (right).

proves that the similarity evaluation is very dependent on the particular person and focuses on different facial aspects, a factor well known from police sketching (Mancusi 2014).

Analyzing the overall similarity scores given by the users (Figure 13) suggests that voters tended to assign scores that were either low or high instead of following the expected normal distribution; they are almost equally divided with the only exception for very low values (range [0,9]). The surprising trend of user-assigned scores motivated a separate test to prove users' consistency over time. After 72 hours, six *evaluators* repeated the score assignment. Interestingly, most votes were similar but some were completely different (50+ difference), with an average difference of above 20.

Constructors who created the composites assigned higher scores than *evaluators* in 75% of the cases, and for the remaining 25% the difference is marginal. Since the high scores assigned by *constructors* are also assigned by some *evaluators*, those people possibly perceived some aspects differently than other users. This suggests that low scores may be due to *constructors* perception rather than a deficiency of the system. In other words, they did not produce a *better* composite because they were already satisfied. Similar cases were found while testing DeepIE, indeed authors often disagreed with the users' choice of the best results (Bontrager and others 2018).

It was expected from expert users – the authors – to achieve the best results, mirroring previous experiments in IEC (Löwe and Risi 2016). Surprisingly, the best composites were generated by novice users, and not all of them even familiar with using complex software (Figure 12).

Discussion and Future Work

The promising results could be improved further by training a GAN with a more extensive and diverse training set. For example, if no celebrities have scars, GANs will probably not become capable of representing scars. The same applies to any unique or not widespread feature.

Feature limitations. Additionally, obtaining feature axes via supervised learning requires significant training data. The lack of labeled face images prevented possible improve-

ID	EXPERT USER	AVG SIMILARITY SCORE	RECOGNITION RATE
GAN 5 GZ 30	N	73.96	100
GAN 5 LZ 20	Y	45.74	100
GAN 5 MT 10	N	29.44	42.86
GAN 5 SZ 30	N	64.44	100
GAN 6 SZ 30	N	87.04	85.71
GAN 6 SC 30	N	59.26	100
GAN 6 AZ 20	N	27.22	85.71
GAN 6 NM 30	N	41.11	57.14
GAN 7 CJ 30	N	55.56	42.86
GAN 7 NZ 30	Y	30.52	14.29
GAN 7 SB 30	N	22.89	42.86
GAN 7 DD 30	N	44	42.86
GAN 9 FM 30	N	16.11	57.14
GAN 9 MT 20	N	63.11	57.14
GAN 9 SZ 30	N	61.85	85.71
GAN 9 LZ 30	Y	37.96	42.86
AVERAGE		47.51	66.07

Figure 12: Each row shows the target image and composite together with an indication if it was created by an expert user. The next column shows the average similarity scores given by the evaluators, followed by the percentage of users that recognized the particular composite correctly.

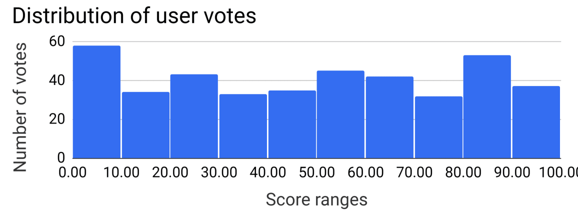

Figure 13: Histogram showing the distribution of similarity scores given by all users for all 16 result composites.

ments and additional axes to be learned. Some workarounds have been attempted to bypass the problem: manual feature research and overlays. In the first case, we tried to infer axes via trial-and-error, with no remarkable results. The second approach involved overlays for altering colors over specific areas, e.g. colored ellipses over the eyes. Their downside is the missing relation with the genotype, so the modifications could not be translated into a latent vector change and re-evolved.

Environmental factors. Forensic artists take into consideration conditions that may affect the perception such as lighting, location and time (Mancusi 2014). It may be useful to train GANs on specific environment-dependent datasets, to further improve our results.

Feature axes improvements. The CelebA (Liu and others 2015) dataset served the purpose of learning 40 features. However, some were marginally relevant and additional ones would be helpful. Some useful additions could be dimensionalities of the face (width, length, distance between eyes) or its shape (oval, round, square). An interesting improvement would be to learn feature axes from computed measures rather than a labeled dataset, e.g. through the usage of facial recognition tools. Measures may include eyes color, emotions, or pose.

Perceptual impact of external features For a similar application, it was demonstrated that blurring external features such as hair with Gaussian filters allows maintaining context while concentrating on more important inner features (Frowd, Bruce, and Hancock 2008). Implementing a similar technique in the CG-GAN system would probably improve its naming rate as for other systems.

Conclusion

The presented composite creation approach suggests the promising involvement of generative techniques in this area. Current composite creation systems mostly belong to two main categories. Some rely on datasets of drawn features that are arranged, moved and stretched to build a face (IQ Biometrix 2003), which are not holistic and only work with pre-defined features. Others evolve parameters to be used by mathematical functions to create a face (Zahradnkov, Duchoviov, and Schreiber 2016; Frowd, Hancock, and Carson 2004; George et al. 2008) but can only create traits that these functions are designed for through expert knowledge. CG-GAN, on the other hand, learns to generate whole faces rather than assembling them from sets of components or mathematical rules. Expert knowledge is not necessary and high-quality colored animated composites are produced by directly learning from pictures. Our results suggest that even inexpert users can produce good composites in a limited time. Most issues appear promisingly simple to solve, granting a likely accuracy improvement. The applied pre-trained generator network is used as is, however it could be re-trained on a more diverse database or different domains.

References

Bontrager, P., et al. 2018. Deep interactive evolution. *Computational Intelligence in Music, Sound, Art and Design Lecture Notes in Computer Science* 267282.

Bontrager, P.; Togelius, J.; and Memon, N. 2017. Deepmasterprint: Generating fingerprints for presentation attacks. *arXiv preprint arXiv:1705.07386*.

Clune, J., and Lipson, H. 2011. Evolving 3d objects with a generative encoding inspired by developmental biology. *ACM SIGEVOlution* 5(4):2–12.

Davis, J.; Solomon, C.; Gibson, S.; and Sulley, L. 2010. A comparison of individual and morphed facial composites created using different systems. In *2010 International Conference on Emerging Security Technologies*.

Frowd, C.; Carson, D.; Ness, H.; Richardson, J.; Morrison, L.; McLanaghan, S.; and Hancock, P. 2005. A forensically valid comparison of facial composite systems. *Psychology Crime and Law* 11.

Frowd, C. D.; Bruce, V.; and Hancock, P. 2008. Changing the face of criminal identification. *The Psychologist* 21.

Frowd, C.; Hancock, P.; and Carson, D. 2004. Evofit: A holistic, evolutionary facial imaging technique for creating composites. *ACM Transactions on Applied Psychology (TAP)* 1.

George, B.; Gibson, S.; I. S. Maylin, M.; and Solomon, C. 2008. Efit-v - interactive evolutionary strategy for the construction of photo-realistic facial composites. In *GECCO'08: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation 2008*, 1485–1490.

González de Prado Salas, P., and Risi, S. 2017. Interactive evolution of complex behaviours through skill encapsulation. In *European Conference on the Applications of Evolutionary Computation*, 853–869. Springer.

Goodfellow, I. J., et al. 2014. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, 2672–2680. Cambridge, MA, USA: MIT Press.

Guan, S. 2018. Generating custom photo-realistic faces using ai. https://github.com/SummitKwan/transparent_latent_gan. [Online; accessed 05-05-2019].

Hoover, A. K.; Szerlip, P. A.; Norton, M. E.; Brindle, T. A.; Merritt, Z.; and Stanley, K. O. 2012. Generating a complete multipart musical composition from a single monophonic melody with functional scaffolding. In *ICCC*, 111–118. Citeseer.

IQ Biometrix. 2003. Faces 4.0. http://www.iqbiometrix.com/products_faces_40.html. [Online; accessed 01-12-2018].

Karras, T., et al. 2017. Progressive growing of gans for improved quality, stability, and variation. *CoRR* abs/1710.10196.

Karras, T.; Laine, S.; and Aila, T. 2018. A style-based generator architecture for generative adversarial networks. *CoRR* abs/1812.04948.

LeCun, Y.; Bengio, Y.; and Hinton, G. 2015. Deep learning. *nature* 521(7553):436.

Liu, Z., et al. 2015. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Löwe, M., and Risi, S. 2016. Accelerating the evolution of cognitive behaviors through human-computer collaboration. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, GECCO '16, 133–140. New York, NY, USA: ACM.

Makhzani, A.; Shlens, J.; Jaitly, N.; and Goodfellow, I. 2016. Adversarial autoencoders. In *International Conference on Learning Representations*. arXiv preprint arXiv:1511.05644.

Mancusi, S. 2014. *Police composite sketch*. Humana.

Nguyen, A.; Yosinski, J.; Bengio, Y.; Dosovitskiy, A.; and Clune, J. 2016. Plug & play generative networks: Conditional iterative generation of images in latent space. *arXiv preprint arXiv:1612.00005*.

Radford, A.; Metz, L.; and Chintala, S. 2016. Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representations (ICLR)*. arXiv preprint arXiv:1511.06434.

Secretan, J.; Beato, N.; D'Ambrosio, D. B.; Rodriguez, A.; Campbell, A.; Folsom-Kovarik, J. T.; and Stanley, K. O. 2011. Picbreeder: A case study in collaborative evolutionary exploration of design space. *Evolutionary Computation* 19(3):373–403.

Simon, J. 2018. Ganbreeder - artbreeder. https://ganbreeder.app/. Accessed: 2018-10-10.

Takagi, H. 2001. Interactive evolutionary computation: Fusion of the capabilities of ec optimization and human evaluation. *Proceedings of the IEEE* 89(9):1275–1296.

Volz, V.; Schrum, J.; Liu, J.; Lucas, S. M.; Smith, A.; and Risi, S. 2018. Evolving mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 221–228. ACM.

Woolley, B. G., and Stanley, K. O. 2011. On the deleterious effects of a priori objectives on evolution and representation. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, GECCO '11, 957–964. New York, NY, USA: ACM.

Zahradnikova, B.; Sutova, Z.; and Schreiber, P. 2017. Interactive evolution of facial composites. *IFAC-PapersOnLine* 50:11776–11781.

Zahradnkov, B.; Duchoviov, S.; and Schreiber, P. 2016. Facial composite systems: review. *Artificial Intelligence Review* 1–22.