Manipulación de Expresiones Faciales vía Espacio Latente de Red Generativa Antagónica (GAN)

Daiana Aranda[1], Julieta Goria[1], Francisco Sandalinas[1], Mateo Suffern[1], and Pablo Negri[1,2][0000−0003−0250−5208]

[1] Departamento de Computación, FCEN-UBA
[2] Instituto de Investigación en Ciencias de la Computación (ICC), UBA-CONICET
pnegri@dc.uba.ar

Resumen StyleGAN [1] destaca como la arquitectura de vanguardia en generación de rostros sintéticos altamente realistas. Su implementación proyecta una imagen en su espacio latente, el cual es posible de manipular por medio de curvas direccionales modificando rasgos de la imagen original. Sin embargo, su alta dimensionalidad provoca que la búsqueda manual de una direccionalidad que produzca un rasgo o gesto dado resulte impracticable. Este trabajo propone una arquitectura neuronal de tipo pseudo-autoencoder que manipula la proyección latente alternando la apariencia del rostro. Esto se realiza gracias a la codificación del gesto facial con los vectores de Action Units. Se consiguió una dinámica de expresiones que permite la transición de un gesto a otro sin necesidad de pasar por el neutral, mejorando la naturalidad de la dinámica gestual.

Keywords: StyleGANv2 · Espacio Latente · Expresiones Faciales.

1. Introducción

En este trabajo, abordaremos la dinámica temporal del retrato como objeto capaz de evolucionar de manera auto-generativa a través del estudio del cambio de apariencia de rostros humanos. Es para esto que utilizaremos imágenes sintéticas creadas por StyleGANv2[1], una red generativa antagónica (GAN) generadora de rostros hiper-realistas. Buscaremos reflejar los cambios de estado de ánimo de la persona en su vida cotidiana en el retrato.

StyleGAN nos permite editar imágenes mediante manipulaciones en su espacio latente asociado. Estos cambios visuales nos permiten cambiar su apariencia o expresión, que puede traducirse como un nuevo estado emocional del modelo.

Los principales trabajos en el área que buscan dominar la dinámica del *manifold* para obtener salidas esperadas del modelo se pueden clasificar de forma amplia al utilizar entrenamientos **no supervisados** y **supervisados**. [1,2,3,4,5,6,7,8,9,10,11]. El entrenamiento no supervisado es principalmente útil en el caso de manipulaciones donde no existen datasets etiquetados o es difícil generarlos [3]. El entrenamiento supervisado utiliza datasets con etiquetas o labels dados por expertos para entrenar, ya sea a otra red o la misma GAN en busca de lograr manipulaciones particulares, como son las expresiones faciales.

Este trabajo propone el uso de una arquitectura neuronal de tipo pseudo-autoencoder con un entrenamiento supervisado que, gracias al uso de vectores de Action Units (AUs) [12], obtenga una transición natural de una expresión facial

sintética a otra. Se presentan también, efectos no deseados y sesgos obtenidos del entrenamiento del pseudo-autoencoder.

2. Codificador de Expresiones

2.1. Espacio Latente

Las redes Neuronales Generativas Antagónicas (GAN) [13] consisten en un sistema donde un discriminador **D** se entrena para maximizar la probabilidad de reconocer entre una imagen real y otra sintética, y un generador **G** que intenta engañarlo, generando imágenes sintéticas realistas a partir de una semilla aleatoria Z. Una vez que la GAN ha sido entrenada, **G** realiza un mapeo no-lineal de esta semilla aleatoria Z para generar una salida de tipo imagen hiper-realista de la forma: $g : Z \to X$.

En el caso de StyleGAN [14] su generador, en vez de aprender un mapeo directo entre Z y X, primero mapea Z a otro Espacio Latente intermedio W. Este espacio está compuesto de 18 canales que codifican información del rostro (género,fondo, color de ojos, etc.) Luego, el W es mapeado a X donde [14] demuestra que el entrelazamiento en la geometría de W es menor respecto del entrelazamiento de la geometría en Z, es decir simplifica el espacio en sí, permitiendo que las modificaciones de un vector latente $\mathbf{w}$ sigan estando en el manifold del modelo.

Este espacio latente aunque no completamente des-entrelazado nos permite que dado $\mathbf{w} \in W$ y una dirección ϕ podemos operar en su vecindad utilizando interpolaciones lineales sucesivas para explorar el espacio en el manifold y observar modificaciones de forma continua en la data $g(\mathbf{w} + (\phi - \mathbf{w}) \cdot \delta)$. Donde δ es un factor de proporcionalidad.

2.2. Arquitectura del pseudo-autoencoder

Se desarrolló un modelo inspirado en la arquitectura de autoencoders. Su objetivo consiste en que, a partir de un vector latente $\mathbf{w}_{in}$ se proyecta un vector latente de salida $\mathbf{w}_{out}$ el cual corresponde a la misma identidad, pose, edad, etc., pero su expresión facial está determinada por un vector de la dinámica de actions units $\mathbf{u}_{in}$. Un ejemplo de esta transición está ejemplificado en la fig. 2. Vamos a ver esta arquitectura en detalle en los párrafos sucesivos.

Los autoencoder son una arquitectura neuronal multicapas cuya salida busca replicar exactamente los datos de entrada. Formalmente, sea $A()$ el modelo autoencoder, tal que $A(\mathbf{x}) = \hat{\mathbf{x}}$ de manera que $\hat{\mathbf{x}} \simeq \mathbf{x}$. Las partes de un autoencoder pueden dividirse en **encoder**, que proyecta la entrada en un espacio latente de menor dimensión, y **decoder** que reconstruye la información inicial sin pérdida de información.

El vector latente $\mathbf{w}_{in}$ es la entrada a la red que se compone de un encoder de dos capas de 512 unidades cada una. A la salida latente se le concatena el vector de la dinámica de las action units $\mathbf{u}_{in}$. Este vector, de 43 elementos, es copiado 5 veces con el objetivo de incrementar las conexiones. Luego, el decoder consiste en 2 capas de 512 neuronas cada una, con la salida final en $\mathbf{w}_{out}$, de manera que $|\mathbf{w}_{in}| = |\mathbf{w}_{out}|$.

El vector de la dinámica de actions units $\mathbf{u}_{in}$ guia la transición del gesto del rostro de la persona. Este vector se obtiene de la siguiente manera. Sea un dataset de fotografías de sujetos realizando distintas expresiones, cada instancia esta compuesta por la dupla $(I_S^i, \mathbf{a}_S^i)$, donde I_S^i es la captura del sujeto i realizando el gesto S, y $\mathbf{a}_S^i$ corresponde al vector de las 43 action units completado manualmente a partir de I_S^i. Cuando una action unit está representada en I_S^i toma valor 1 en el vector. En caso contrario toma valor 0. Luego, como el objetivo es que el pseudo-autoencoder encuentre la transición de $\mathbf{w}_{in}$ a $\mathbf{w}_{out}$ precisamos la información de aquellas novedades respecto de los vectores de action units que corresponden a cada fotografía. La transición de un gesto a S hacia T, el vector dinámico se obtendría: $\mathbf{u}_{in}^{S-T} = \mathbf{a}_T^i - \mathbf{a}_S^i$.

Cada elemento k en $\mathbf{u}_{in}^{S-T}$, corresponde a la action unit en la posición k de $\mathbf{a}_T^i$ y $\mathbf{a}_S^i$ tomando un valor:

$$\mathbf{u}_{in}[k] = \begin{cases} 1 & \text{si } \mathbf{a}_T^i[k] \text{ esta activada y no lo está en } \mathbf{a}_S^i \\ 0 & \text{si no hay cambio en esta action unit entre } \mathbf{a}_S^i[k] \text{ y } \mathbf{a}_T^i[k] \\ -1 & \text{si esta action unit se desactiva en } \mathbf{a}_T^i[k] \end{cases}$$

3. Experimentos y Resultados

3.1. Dataset de Entrenamiento

El entrenamiento del pseudo-autoencoder precisa un dataset de entrenamiento con N gestos por persona y su correspondiente action unit.

El dataset OSU [15] captura 230 sujetos realizando 22 gestos pre-establecidos, además posee las action units de cada imagen en un formato binario (0 no-detectada, 1 detectada).

Utilizamos un pre tratamiento donde cada imagen de rostro se alineo usando MTCNN [16] a una posición compatible con el modelo pre-entrenado StyleGANv2.Luego se obtuvieron todos los vectores latentes $\mathbf{w}_i^S$ de cada imagen, que sumado al vector de action units $\mathbf{a}_S^i$ conforman la tupla de dato del sujeto i realizando el gesto S.

3.2. Entrenamiento del pseudo-Autoencoder

El set de entrenamiento está compuesto por las entradas X y las salidas esperadas Y, calculados utilizando el set OSU y sus etiquetas. El set de entrada $X = \{(\mathbf{w}_1^1, \mathbf{u}_1^{1-2}), (\mathbf{w}_1^1, \mathbf{u}_1^{1-3}), (\mathbf{w}_1^1, \mathbf{u}_1^{1-4}), ...(\mathbf{w}_{230}^{22}, \mathbf{u}_{230}^{22-21})\}$ y los valores esperados son, respectivamente, $Y = \{\mathbf{w}_1^2, \mathbf{w}_1^3, \mathbf{w}_1^4, ..., \mathbf{w}_{230}^{21}\}$. Entonces, la muestra j de $X(j) = (\mathbf{w}_i^S, \mathbf{u}_i^{S-T})$ y su target es $Y(j) = \mathbf{w}_i^T$, que significa que ante la entrada del vector latente ($\mathbf{w}_i^S$ con la expresión S, la salida tiene que tener la expresión en el vector latente $\mathbf{w}_i^T$. Esto define la función de pérdida como $\mathcal{L}_j = \| \mathbf{w}_i^T - \hat{\mathbf{w}}_i^T \|$, donde $\hat{\mathbf{w}}_i^T$ es el vector latente estimado por el pseudo-autoencoder teniendo como entrada $X(j)$.

El modelo se entrenó durante 150 épocas usando un optimizador Adam [17] con un *learning rate* igual a $10e^{-3}$ y un *weigth decay* de $10e^{-5}$. Además, se estableció un decaimiento del learning rate con un factor $\gamma = 0{,}5$ cada 10 épocas, para un aprendizaje suave hacia las épocas superiores.

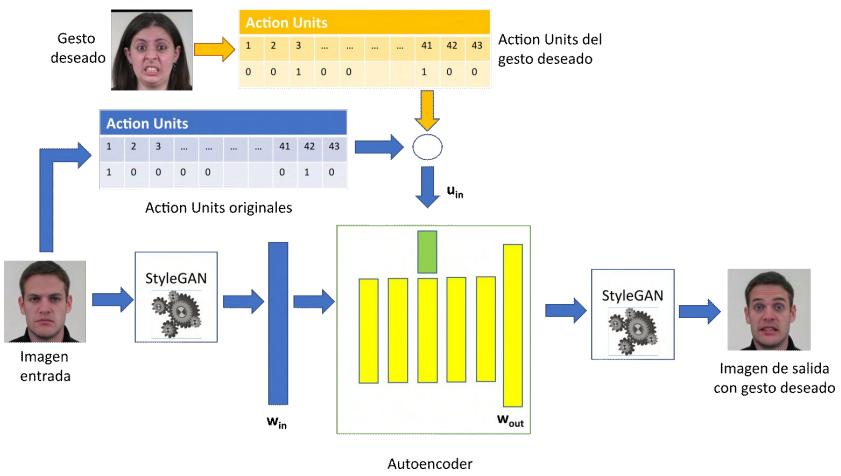

Figura 1: Funcionamiento del pseudo-autoencoder: Una imagen de entrada con su vector de AUs, es proyectado por el modelo a un vector latente que posee la expresión deseada, definida por las AUs de la imagen de la mujer

3.3. Transición dinámica y generación de vídeos

Figura 2: Transición de una expresión a otra dada por una interpolación lineal entre dos vectores latentes.

Dada una imagen de entrada, cuya proyeccion llamaremos $\mathbf{w}_s$, y un vector de AUs que describen el gesto deseado $\mathbf{g}_t$ podemos generar, gracias al pseudo-autoencoder, el $\mathbf{w}_t$ asociado a $\mathbf{g}_t$. Sabemos que las imagenes cambian linealmente si interpolamos linealmente entre dos espacios latentes[14], tambien sabemos que se cumple $\mathbf{v} = \mathbf{w}_t - \mathbf{w}_s$ donde $\mathbf{v}$ es la direccion que te lleva de $\mathbf{w}_s$ a $\mathbf{w}_t$. Esto nos permite utilizar la propiedad de que $\mathbf{w}_i = \mathbf{w}_s + \mathbf{v} * \alpha$, $\alpha \in [0, 1]$, donde $\mathbf{w}_i = \mathbf{w}_s \iff \alpha = 0 \wedge \mathbf{w}_i = \mathbf{w}_t \iff \alpha = 1$.

Veamos que modulando este parámetro α podemos generar imágenes *intermedias* entre el target y la imagen inicial. Como ejemplo, en la fig. 2 partimos

de 3 imagenes del dataset OSU con sus respectivos $\mathbf{w}_s$, donde para cada $\mathbf{w}_s$ se quiere llegar al mismo gesto($\mathbf{g}_{contento}$ que tiene asociado su $\mathbf{w}_c$). Utilizamos el psedoautoencoder con sus respectivos AU's para obtener el $\mathbf{w}_c$. La transicion de imagenes de una expresion a otra se logra mediante $\mathbf{w}_i = \mathbf{w}_s + (\mathbf{w}_c - \mathbf{w}_s) * \alpha$, $\alpha \in \{0, 0{,}25, 0{,}50, 0{,}75, 1\}$con $\mathbf{w}_i$ que es el vector asociado a cada imagen intermedio.

3.4. Bias

Uno de los problemas que se nos presentó al generar los rostros fue la pérdida de identidad del rostro *target* luego de pasar por el pseudo-autoencoder. Si bien el proceso funciona muy bien para la modificación de gestos, el rostro pierde muchos de los factores lo identifican, como muestra la fig. 3. Esto se debe a que el dataset de entrenamiento solo contiene los rostros y expresiones de 230 personas, por lo tanto la imagen resultante imita los rasgos de dicho grupo.

Figura 3: Bias del pseudo-autoencoder al proyectar imágenes reales fuera de OSU.

4. Conclusiones

Este trabajo propone un método para la generación de imágenes sintéticas correspondientes a una secuencia de gestos naturales, los cuales podemos asociar al estado de ánimo de la persona. Las redes GAN se muestran como una herramienta completa, sobre la cual se puede construir la obra y darle diferentes improntas. Quedan como perspectivas varios puntos que fuimos dilucidando en el desarrollo de la demo como la incorporación de otras manipulaciones, movimientos de pose o alteraciones en la edad. que permitirían enriquecer el producto final. Por otro lado, los sesgos que detectamos a la hora de proyectar una identidad por el autoencoder nos demuestra la importancia de manejar con cuidado los datos de entrada.

Referencias

1. T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *CVPR*, pp. 8110–8119, 2020.
2. E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, "Ganspace: Discovering interpretable gan controls," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9841–9850, 2020.
3. L. Goetschalckx, A. Andonian, A. Oliva, and P. Isola, "Ganalyze: Toward visual definitions of cognitive image properties," in *ICCV*, pp. 5744–5753, 2019.
4. A. Khosla, A. S. Raju, A. Torralba, and A. Oliva, "Understanding and predicting image memorability at a large scale," in *ICCV*, pp. 2390–2398, 2015.
5. A. Voynov and A. Babenko, "Unsupervised discovery of interpretable directions in the gan latent space," in *International conference on machine learning*, pp. 9786–9796, PMLR, 2020.
6. C. Tzelepis, G. Tzimiropoulos, and I. Patras, "Warpedganspace: Finding non-linear rbf paths in gan latent space," in *ICCV*, pp. 6393–6402, 2021.
7. X. Yao, A. Newson, Y. Gousseau, and P. Hellier, "A latent transformer for disentangled face editing in images and videos," in *ICCV*, pp. 13789–13798, 2021.
8. T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.
9. Y. Fan, F. Tian, X. Tan, and H. Cheng, "Facial expression animation through action units transfer in latent space," *Computer Animation and Virtual Worlds*, vol. 31, no. 4-5, p. e1946, 2020.
10. Y. Viazovetskyi, V. Ivashkin, and E. Kashin, "Stylegan2 distillation for feed-forward image manipulation," in *ECCV*, pp. 170–186, Springer, 2020.
11. T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," in *CVPR*, pp. 8798–8807, 2018.
12. P. Ekman and W. V. Friesen, "Facial action coding system," *Environmental Psychology & Nonverbal Behavior*, 1978.
13. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.
14. T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *CVPR*, pp. 4401–4410, 2019.
15. S. Du, Y. Tao, and A. M. Martinez, "Compound facial expressions of emotion," *Proceedings of the National Academy of Sciences*, vol. 111, no. 15, pp. E1454–E1462, 2014.
16. K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE signal processing letters*, vol. 23, no. 10, pp. 1499–1503, 2016.
17. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization. iclr. 2015," *arXiv preprint arXiv:1412.6980*, vol. 9, 2015.