

# Localización de landmarks cefalométricos por medio de técnicas de few-shot learning y análisis de redes convolucionales

Alejandro Borrego Megías

**Tutores:** Pablo Mesejo Santiago, Javier Merí de la Maza

**Mentor:** Guillermo Gómez Trenado

Doble Grado en Ingeniería Informática y Matemáticas

Universidad de Granada, España

27 de Noviembre de 2022



UNIVERSIDAD  
DE GRANADA

# Índice Primera Parte

## Análisis de redes convolucionales

① Introducción

② Modelización

③ Invarianza por traslaciones



# Redes Neuronales Convolucionales

- Buen rendimiento, comprobable empíricamente.
- Vía de estudio abierta en lo que se refiere a la modelización matemática y la justificación teórica de estos resultados.

Propiedades:

Invarianza por traslaciones



Invarianza frente a pequeñas deformaciones



# Invarianza por traslaciones

Trabajamos sobre el espacio de funciones  $L^2(\mathbb{R}^d)$ .

## Definición de traslación

Sea  $f \in L^2(\mathbb{R}^d)$ ,  $L_c f(x) = f(x - c)$  es la traslación de  $f$  por  $c \in \mathbb{R}^d$ .

## Invarianza por traslaciones de un operador $\Phi$

Decimos que un operador  $\Phi$  sobre  $L^2(\mathbb{R}^d)$ , es invariante por traslaciones si  $\Phi(L_c f) = \Phi(f)$  para todo  $f \in L^2(\mathbb{R}^d)$  y para todo  $c \in \mathbb{R}^d$ .



# Invarianza frente a pequeñas deformaciones

*Deformación  $\implies$  Difeomorfismo*

*Deformaciones pequeñas  $\implies$  Difeomorfismos cercanos a traslaciones*

## Definición

Denotemos  $L_\tau f(x) = f(x - \tau(x))$  como la acción del difeomorfismo  $1 - \tau$  sobre  $f$ .

Donde  $\tau$  es el campo de desplazamiento.



# Invarianza frente a pequeñas deformaciones

Invarianza frente a pequeñas deformaciones



Lipschitz-continuidad frente a la acción de difeomorfismos

Invarianza frente a pequeñas deformaciones de un operador  $\Phi$

$$\|\Phi(f) - \Phi(L_\tau f)\| \leq C\|f\|(\|\nabla\tau\|_\infty + \|H\tau\|_\infty).$$

Con  $f \in L^2(\mathbb{R}^d)$  y  $C > 0$ .

## Objetivo

**Encontrar un operador  $\Phi$  que cumpla lo anterior y modelice una red convolucional**



# Módulo de la transformada de Fourier

$$\Phi(f) = |\widehat{f}| \quad f \in L^2(\mathbb{R}^2)$$

Con este operador observamos que:

- Es invariante por traslaciones.
- No es Lipschitz continuo frente a pequeñas deformaciones.

Debemos buscar otro operador.



## Alternativa: Ondeletas

Usaremos ondeletas madre del tipo:

$$\psi(x) = e^{i\eta x} \Theta(x)$$

donde  $\Theta(x)$  es una función real con soporte en una bola de baja frecuencia en  $x = 0$ , cuyo radio es del orden de  $\pi$ .

### Escalado y rotación de ondeleta madre

Una ondeleta madre escalada por un factor  $2^j$  con  $j \in \mathbb{Z}$  y rotada por  $r \in G$  siendo  $G$  el grupo finito de rotaciones, se escribe:

$$\psi_{2^j r}(x) = 2^j \psi(2^j r^{-1} x).$$



# La transformada de Littlewood-Paley

Con esta generamos la siguiente base ortonormal de ondeletas:

$$\{\psi_\lambda(x)\}_{\lambda=2^j r \in 2^{\mathbb{Z}} \times G}$$

## Transformada de Littlewood-Paley

$$\forall x \in \mathbb{R}^d \quad W[\lambda]f(x) = f * \psi_\lambda(x) = \int f(u)\psi_\lambda(x-u)du.$$

donde  $\lambda \in 2^j r \in 2^{\mathbb{Z}} \times G$

El operador  $W[\lambda]f = f * \psi_\lambda$  es Lipschitz-continuo bajo la acción de difeomorfismos para  $f \in L^2(\mathbb{R}^d)$  fijo.

¿Pero invariante a traslaciones?



## Problema: La escala



Figure: Imagen extraída de <sup>1</sup>.

Fijada una escala  $2^J$   $J \in \mathbb{Z}$ , se establece un umbral tal que solo se mantienen las ondeletas de escala  $2^j > 2^{-J}$ .

---

<sup>1</sup>M. Works (n.d.). *Continuous Wavelet Transform and Scale-Based Analysis*.  
[=https://www.mathworks.com/help/wavelet/gs/continuous-wavelet-transform-and-scale-based-analysis.html](https://www.mathworks.com/help/wavelet/gs/continuous-wavelet-transform-and-scale-based-analysis.html),

## Problema: La escala

Surge la necesidad de promediar las frecuencias no cubiertas por el factor de escala fijado:

$$A_J f = f * \phi_{2^J} \text{ con } \phi_{2^J}(x) = 2^{-J} \phi(2^{-J}x).$$

Así, los coeficientes obtenidos, fijada una escala son:

$$W_J f = \{A_J f, (W[\lambda]f)_{\lambda \in \Lambda_J}\}$$

$$\text{con } \Lambda_J = \{\lambda = 2^j r : r \in G^+, 2^j > 2^{-J}\}.$$

- Bajo hipótesis no muy restrictivas sobre las ondeletas y  $\phi$  los coeficientes cumplen  $\|W_J\| = 1$ .



# Restricciones

Trabajaremos con funciones reales.

- $W_J$  es unitario.
- $\widehat{\phi}(\omega)$  es real y simétrica, por lo que  $\phi$  también lo será.
- Las derivadas de  $\phi$  pertenecen a  $L^1(\mathbb{R}^d)$ .



# Propagador de dispersión

Condición para coeficientes I.T.

Si  $U[\lambda]$  es un operador definido en  $L^2(\mathbb{R}^d)$ , no necesariamente lineal pero que conmuta con traslaciones, entonces  $\int_{\mathbb{R}^d} U[\lambda]f(x)dx$  es invariante a traslaciones si es finito.

Pero como  $\int_{\mathbb{R}^d} \psi(x)dx = 0 \implies \int_{\mathbb{R}^d} f * \psi(x)dx = 0$ .



# Propagador de dispersión

Para obtener coeficientes invariantes por traslaciones.

$$U[\lambda]f = M[\lambda]W[\lambda]f$$

El operador más sencillo que garantiza coeficientes invariantes por traslaciones y Lipschitz-continuidad frente a difeomorfismos es:

Definición del operador  $U[\lambda]$

$$U[\lambda]f = M[\lambda]W[\lambda]f = |f * \psi_\lambda| = \left| \int_{\mathbb{R}^d} f(u)\psi_\lambda(x-u)du \right|$$



# Propagador de dispersión

## Camino

Una secuencia ordenada  $p = (\lambda_1, \lambda_2, \dots, \lambda_m)$  con  $\lambda_k \in \Lambda_\infty = 2^{\mathbb{Z}} \times G^+$  se denomina **camino**. Al camino vacío se le denota por  $p = \emptyset$ .

Así, sobre un camino  $p$  definimos el **propagador de dispersión** como:

$$U[p]f = U[\lambda_m] \dots U[\lambda_2]U[\lambda_1]f = | | f * \psi_{\lambda_1} | * \psi_{\lambda_2} | \dots | * \psi_{\lambda_m} |$$

Además:

$$\int_{\mathbb{R}^d} U[p]f(x)dx < \infty$$



# Operador de ventana

## Definición operador de ventana

Sea  $J \in \mathbb{Z}$  y  $\mathcal{P}_J$  el conjunto de caminos finitos  $p = (\lambda_1, \lambda_2, \dots, \lambda_m)$  con  $\lambda_k \in \Lambda_J$  y  $|\lambda_k| = 2^k > 2^{-J}$ . Una ventana de transformada de dispersión se define para todo  $p \in \mathcal{P}_J$  por

$$S_J[p]f(x) = U[p]f * \phi_{2^J}(x) = \int_{\mathbb{R}^d} U[p]f(u)\phi_{2^J}(x-u)du.$$

$$S_J[p]f(x) = |f * \psi_{\lambda_1}| * \psi_{\lambda_2} | \dots | * \psi_{\lambda_m} | * \phi_{2^J}(x).$$

# Diferencias y Similitudes con una red convolucional

## Similitudes:

- Cascada de convoluciones (operador  $W[\lambda]$ ).
- Capas de *pooling* (operador  $M[\lambda]$  y  $\phi_{2^J}$ ).
- Si  $p$  tiene longitud  $m$ :  $S_J[p]f(x)$  equivale al resultado de la capa  $m$  de la red.

## Diferencias:

- Los pesos no se aprenden.



# Ondeletas admisibles

## ondeletas admisibles

Una ondeleta de dispersión se dice que es admisible si existe  $\eta \in \mathbb{R}^d$  y una función  $\rho \geq 0$ , con  $|\widehat{\rho}(\omega)| \leq |\widehat{\phi}(2\omega)|$  y  $\widehat{\rho}(0) = 1$ , tal que la función:

$$\widehat{\Psi}(\omega) = |\widehat{\rho}(\omega - \eta)|^2 - \sum_{k=1}^{+\infty} k(1 - |\widehat{\rho}(2^{-k}(\omega - \eta))|^2)$$

satisface:

$$\alpha = \inf_{1 \leq |w| \leq 2} \sum_{j=-\infty}^{\infty} \sum_{r \in G} \widehat{\Psi}(2^{-j} r^{-1} \omega) |\widehat{\psi}(2^{-j} r^{-1} \omega)|^2 > 0.$$



# No-expansividad y conservación de la norma

## No-expansividad

Para  $f, h \in L^2(\mathbb{R}^d)$  y  $J \in \mathbb{Z}$  se cumple

$$\|S_{J+1}[\mathcal{P}_{J+1}]f - S_{J+1}[\mathcal{P}_{J+1}]h\| \leq \|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]h\|.$$

## Conservación de la norma

Si las ondeletas son admisibles, entonces para toda  $f \in L^2(\mathbb{R}^d)$  se tiene que

$$\lim_{m \rightarrow \infty} \sum_{n=m}^{\infty} \|S_J[\Lambda_J^n]f\|^2 = 0$$

y

$$\|S_J[\mathcal{P}_J]f\| = \|f\|.$$



# Invarianza por traslaciones

## Invarianza por traslaciones

Para ondeletas de dispersión admisibles se tiene que

$$\forall f \in L^2(\mathbb{R}^d), \quad \forall c \in \mathbb{R}^d \quad \lim_{J \rightarrow \infty} \|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]L_c f\| = 0.$$



## Índice Segunda Parte

### Localización de landmarks cefalométricos por medio de técnicas de few-shot learning

- ④ Definición del Problema
- ⑤ Fundamentos Teóricos
- ⑥ Estado del arte
- ⑦ Experimentos
- ⑧ Conclusiones



## Landmarks cefalométricos

- Puntos de referencia que **definen la morfología del cráneo**.
  - **Correspondencia** directa **cráneo-cara**.
  - Se emplean en **Antropología forense**.

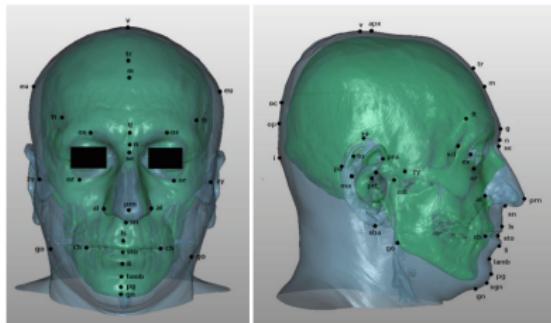


Figure: Imagen extraída de <sup>2</sup>

<sup>2</sup>S. Damas, O. Cordón, and O. Ibáñez (2020). *Handbook on craniofacial superimposition: The MEPROCS project*. Springer Nature

## Landmarks faciales

- Puntos de referencia **marcados sobre la cara**.
  - **No guardan relación** con la morfología del cráneo.
  - Se emplean para **reconocimiento facial**.

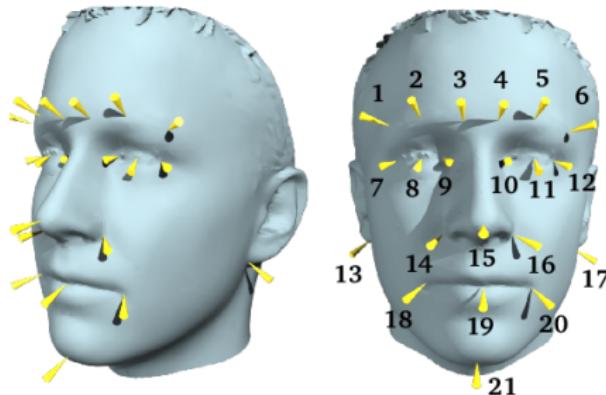


Figure: Landmarks marcados en AFLW<sup>3</sup>

<sup>3</sup>M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof (2011). "Annotated Facial Landmarks in the Wild: A large-scale, real-world database for facial landmark localization". In: *2011 IEEE International Conference on Computer Vision*.

## Base de datos

- 167 imágenes **in-the-wild**.
  - Imágenes a **color** y escala de grises.
  - **30 landmarks** a predecir.



## Objetivos

- ① Revisión del **estado del arte**.
  - ② Investigación sobre los **Autoencoders y redes adversarias** existentes.
  - ③ Estudio y **preprocesamiento** de la base de datos.
  - ④ Estudio **experimental** usando diversos modelos que derivan del framework principal.



# Arquitectura de una red neuronal convolucional

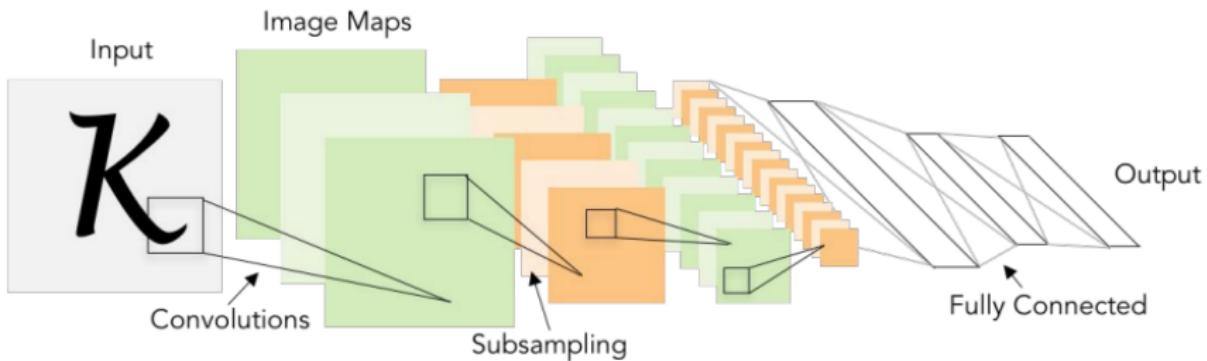


Figure: Imagen extraída de <sup>4</sup>.

---

<sup>4</sup>Fei-Fei Li, Justin Johnson, Serena Young , Stanford University (2017). *cs231n*. =  
<http://cs231n.stanford.edu/2017/syllabus.html>,

# Adversarial Autoencoder (AAE)

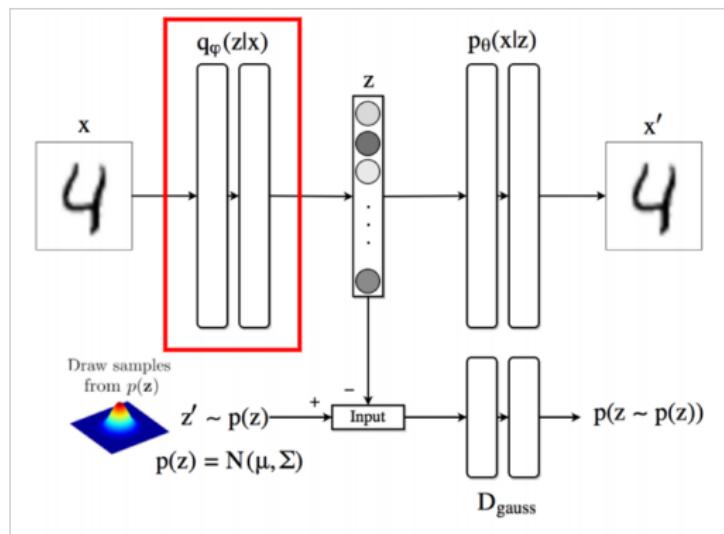


Figure: Imagen extraída de <sup>5</sup>.

<sup>5</sup>F. Rashad (2020). *Adversarial Auto Encoder (AAE)*.  
<https://medium.com/vitrox-publication/adversarial-auto-encoder-AAE-a3fc86f71758>,

# Adversarial Autoencoder (AAE)

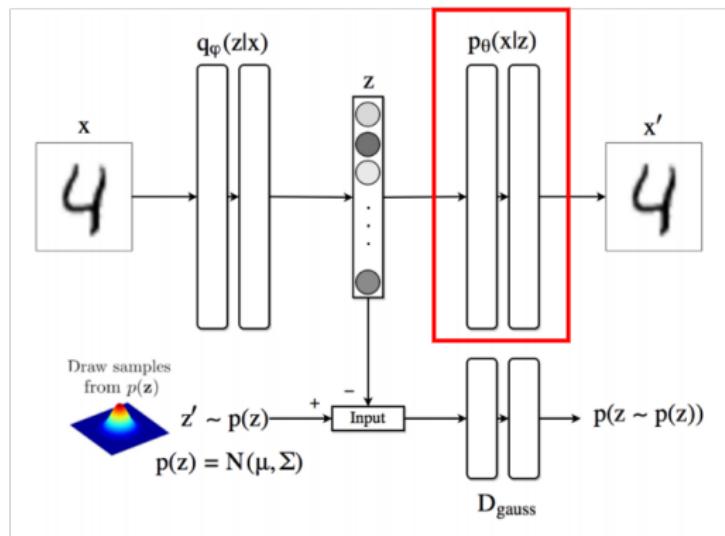


Figure: Imagen extraída de <sup>6</sup>.

<sup>6</sup> F. Rashad (2020). *Adversarial Auto Encoder (AAE)*.  
[=https://medium.com/vitrox-publication/adversarial-auto-encoder-AAE-a3fc86f71758](https://medium.com/vitrox-publication/adversarial-auto-encoder-AAE-a3fc86f71758),

# Adversarial Autoencoder (AAE)

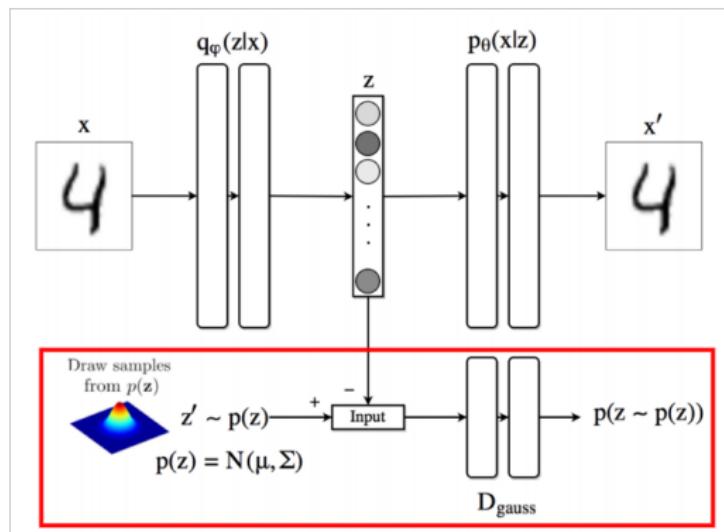


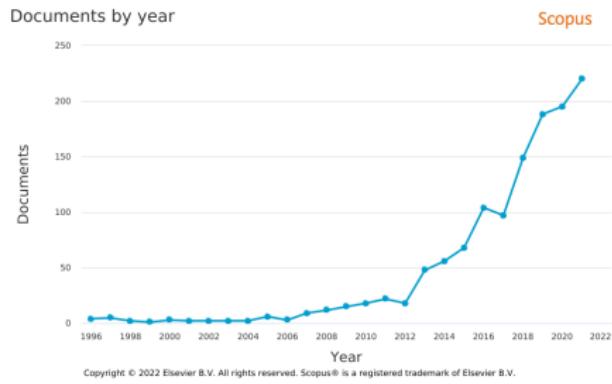
Figure: Imagen extraída de <sup>7</sup>.

<sup>7</sup> F. Rashad (2020). *Adversarial Auto Encoder (AAE)*.  
<https://medium.com/vitrox-publication/adversarial-auto-encoder-AAE-a3fc86f71758>,

## Búsqueda Scopus

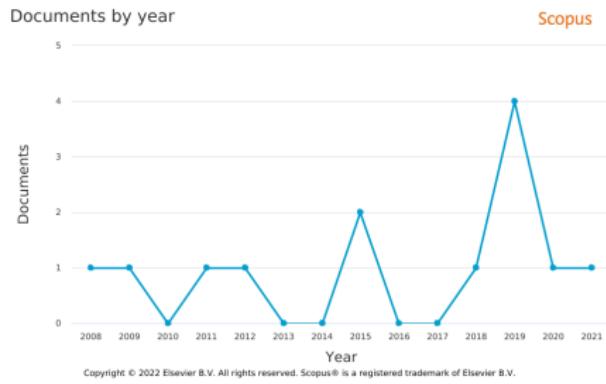
## Landmarks faciales:

- Gran número de artículos.
  - Deep learning.



### **Landmarks cefalométricos:**

- Escasa literatura.
  - Gran variedad de enfoques.



## Estado del arte

- TFG Realizado por Guillermo Gómez <sup>8</sup>.
- Expuesto en Conferencia Forense en 2019.
- Modelo basado en **HyperFace-ResNet**.
- **Predicción y clasificación** de landmarks cefalométricos.
- **Misma base de datos** + 99 modelos 3D.
- Se disponía del **código** y los **resultados**.

---

<sup>8</sup> G. G. Trenado (2019). "Aprendizaje Profundo y Antropología Forense". In: *Universidad de Granada*

# 3FabRec

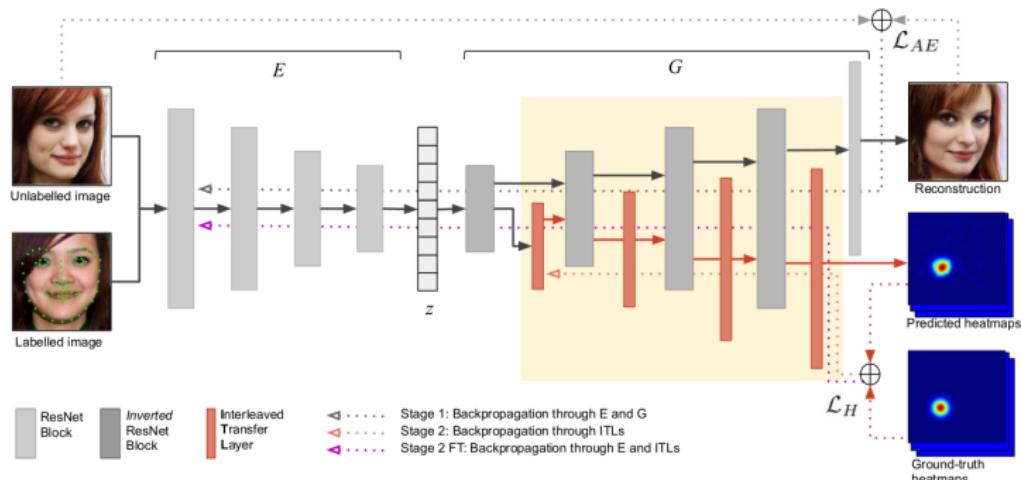


Figure: Imagen extraída de <sup>9</sup>.

<sup>9</sup> B. Browatzki and C. Wallraven (2020). "3fabrec: Fast few-shot face alignment by reconstruction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6110–6120

# 3FabRec

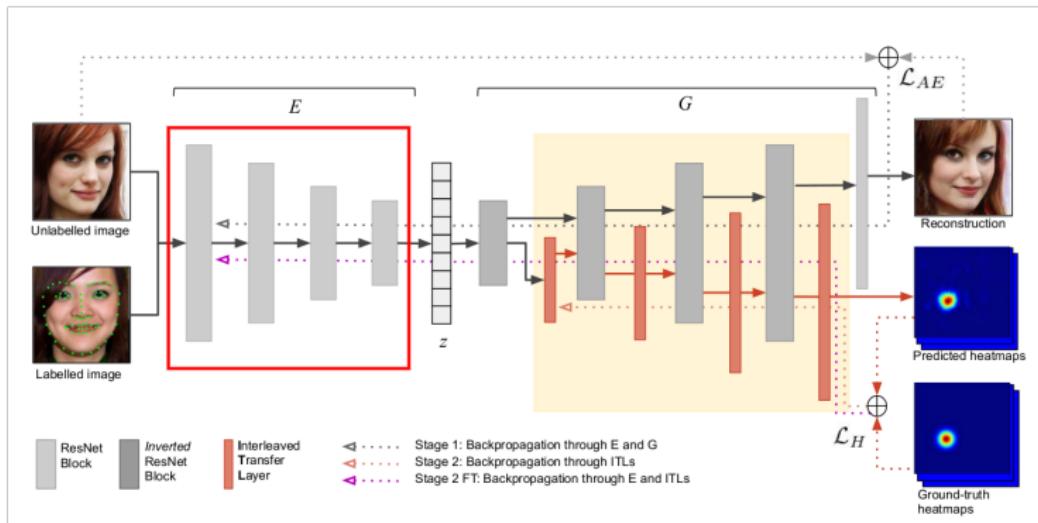


Figure: Imagen extraída de <sup>10</sup>.

<sup>10</sup> B. Browatzki and C. Wallraven (2020). "3fabrec: Fast few-shot face alignment by reconstruction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6110–6120

# 3FabRec

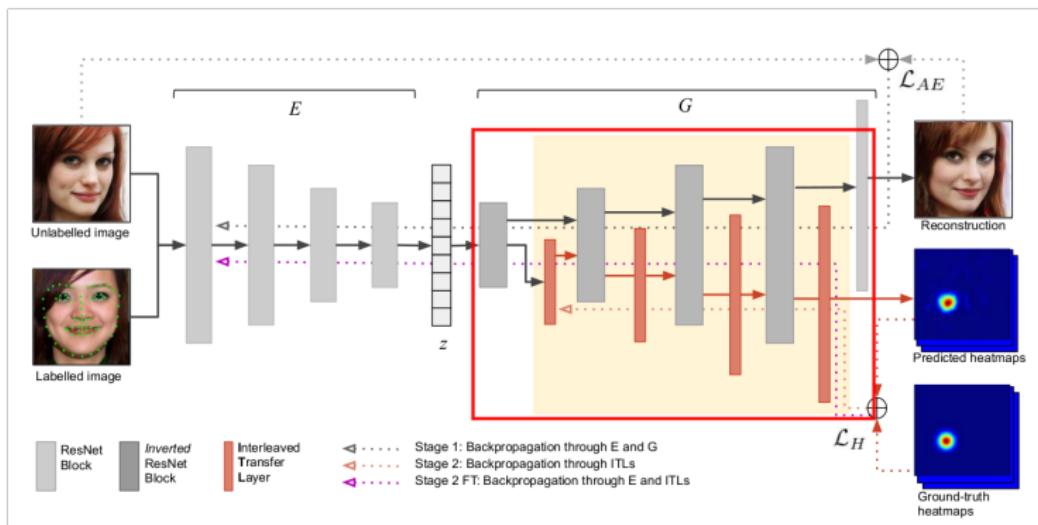


Figure: Imagen extraída de <sup>11</sup>.

<sup>11</sup> B. Browatzki and C. Wallraven (2020). “3fabrec: Fast few-shot face alignment by reconstruction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6110–6120

# 3FabRec

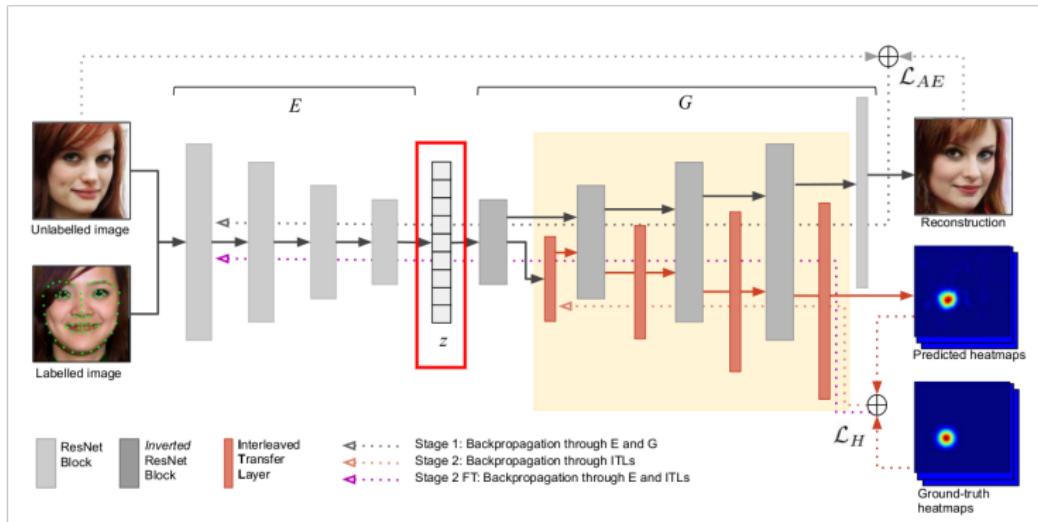


Figure: Imagen extraída de <sup>12</sup>.

<sup>12</sup> B. Browatzki and C. Wallraven (2020). "3fabrec: Fast few-shot face alignment by reconstruction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6110–6120

# 3FabRec

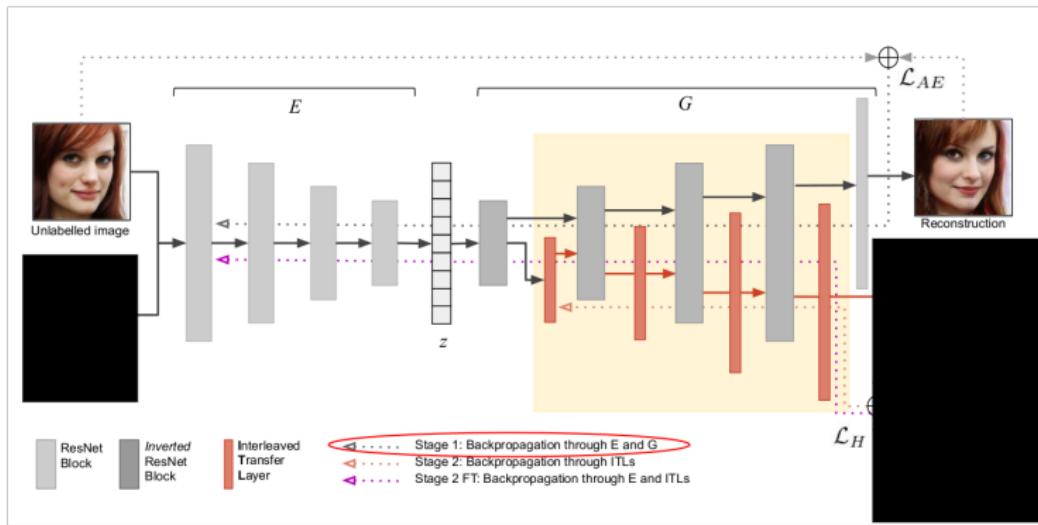


Figure: Imagen extraída de <sup>13</sup>.

<sup>13</sup> B. Browatzki and C. Wallraven (2020). "3fabrec: Fast few-shot face alignment by reconstruction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6110–6120

# 3FabRec

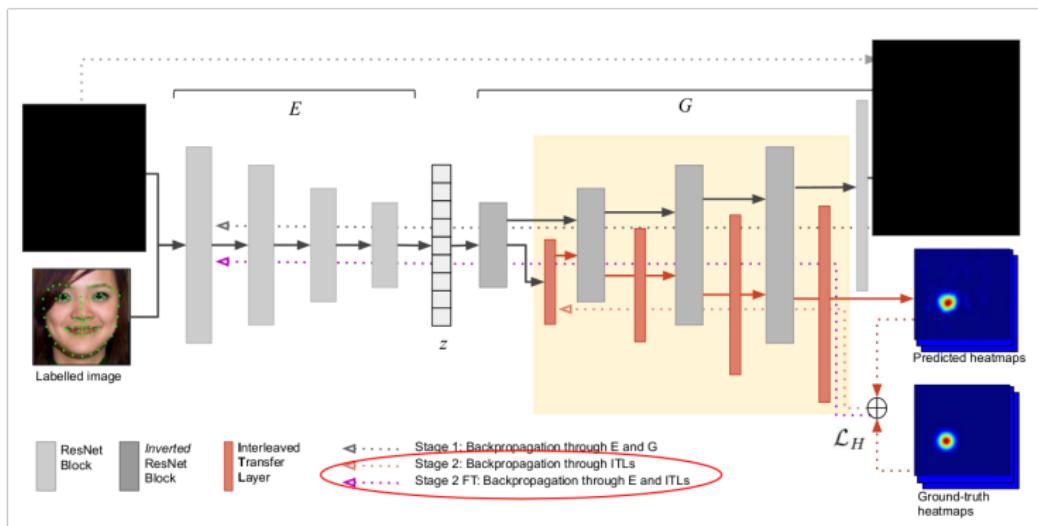


Figure: Imagen extraída de <sup>14</sup>.

<sup>14</sup> B. Browatzki and C. Wallraven (2020). "3fabrec: Fast few-shot face alignment by reconstruction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6110–6120

# 3FabRec

## Bases de datos de la parte no supervisada

- VGGFace2 + AffectNet
- En total 2.1 millones de imágenes.

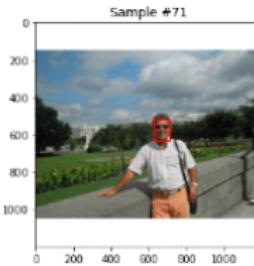
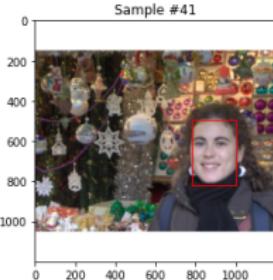
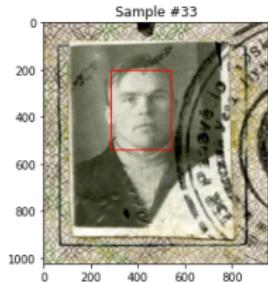
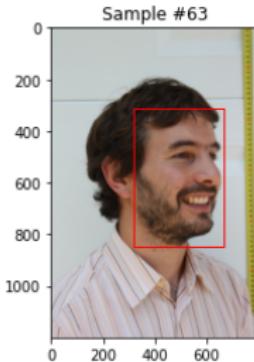
## Bases de datos de la parte supervisada

- 300W = 3.148 imágenes.
- AFLW = 24.386 imágenes.
- WFLW = 10.000 imágenes.



# Preprocesamiento

- Realizamos **detección + cropping** de caras en las imágenes del dataset.
- Empleamos red auxiliar **FaceNet**<sup>15</sup>.
- **Problema:** Algunos landmarks quedan fuera del *bounding box*. Se necesita un **reajuste**.




---

<sup>15</sup>F. Schroff, D. Kalenichenko, and J. Philbin (2015). "FaceNet: A Unified Embedding for Face Recognition and Clustering". In: CoRR abs/1503.03832. arXiv: 1503.03832

# Reajuste 3FabRec

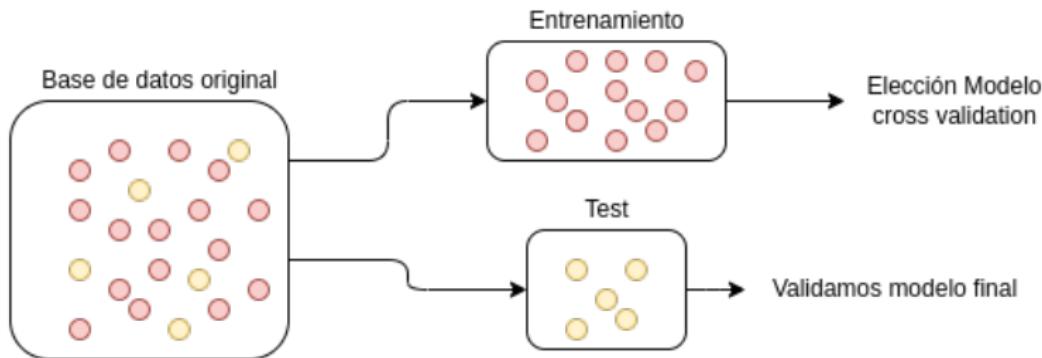
- Bounding box cuadrado de lado  $\max(h, w)$ .
- Ampliación uniforme por el factor  $\frac{crop\_size+margin}{crop\_size}$ .
- Reescalado a **tamaño**  $256 \times 256$ .
- **Importante** para que los datos sigan distribución similar a la usada para entrenar 3FabRec.



## Partición de los datos

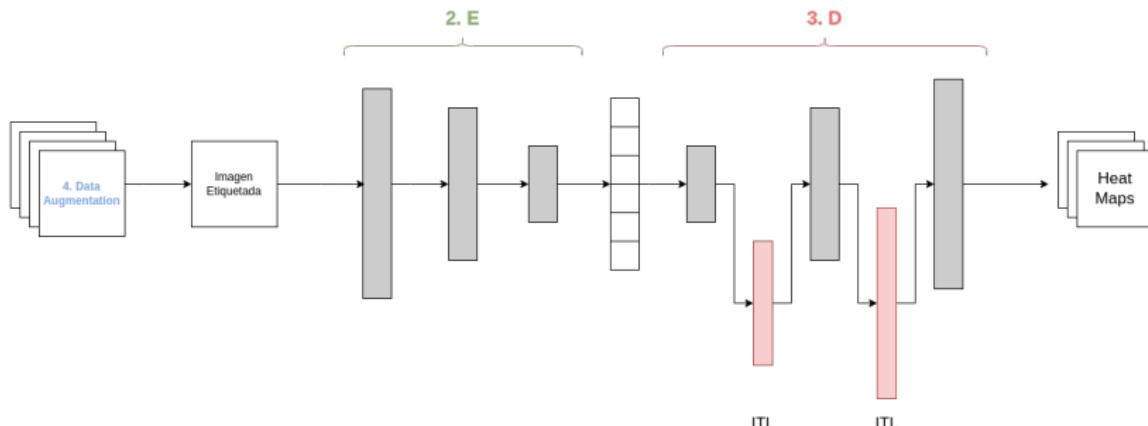
Se dividen los datos en entrenamiento y test:

- **131** imágenes para entrenamiento (80%). Usaremos **cross-validation 5 fold**.
- **32** imágenes para test (20%).
- Durante el preprocessamiento descartamos 4 imágenes.



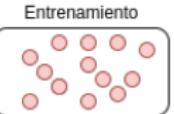
# Experimentos

- **Modelo Base:** entrenamiento ITLs → **Adam**
- **Ajuste fino del Encoder:** entrenamiento ITLs + Encoder → **Adam**
- **Ajuste fino del Decoder:** entrenamiento ITLs + Decoder → **Adam**
- **Data Augmentation:** entrenamiento ITLs con técnicas de Data Augmentation → **Adam**



# Validación de los modelos

## Cross-Validation 5-fold.



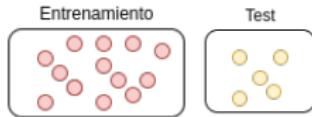
Iteración 1	Test	Entrenamiento	Entrenamiento	Entrenamiento	Entrenamiento
Iteración 2	Entrenamiento	Test	Entrenamiento	Entrenamiento	Entrenamiento
Iteración 3	Entrenamiento	Entrenamiento	Test	Entrenamiento	Entrenamiento
Iteración 4	Entrenamiento	Entrenamiento	Entrenamiento	Test	Entrenamiento
Iteración 5	Entrenamiento	Entrenamiento	Entrenamiento	Entrenamiento	Test

# Métricas empleadas

## Métricas entrenamiento

- MSE:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$



## Métricas Validación

- Error de reconstrucción:

$$\text{Reconstruction Loss} = \frac{\sum_{i=1}^n |p_i - q_i|}{N}$$

- NME:

$$NME = \frac{\sqrt{\sum_{i=1}^n (p_{i,1} - q_{i,1})^2 + (p_{i,2} - q_{i,2})^2}}{N}$$

- SSIM:

$$cs(x, y) = \frac{1}{|w|} \sum c(x_w, y_w) s(x_w, y_w)_w$$

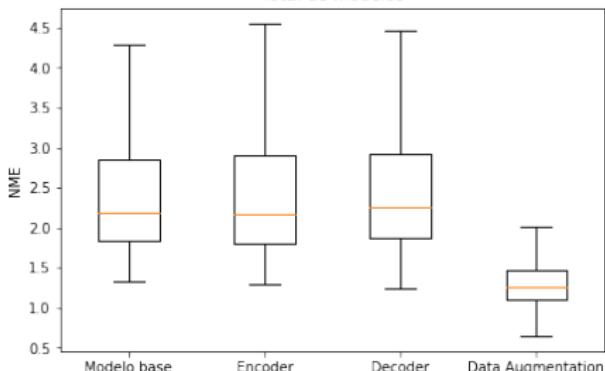
# Análisis Cuantitativo

Para cada imagen:

- NME medio.
- Error de reconstrucción medio.

Se concatenan las salidas de **cross validation**:

Total de modelos

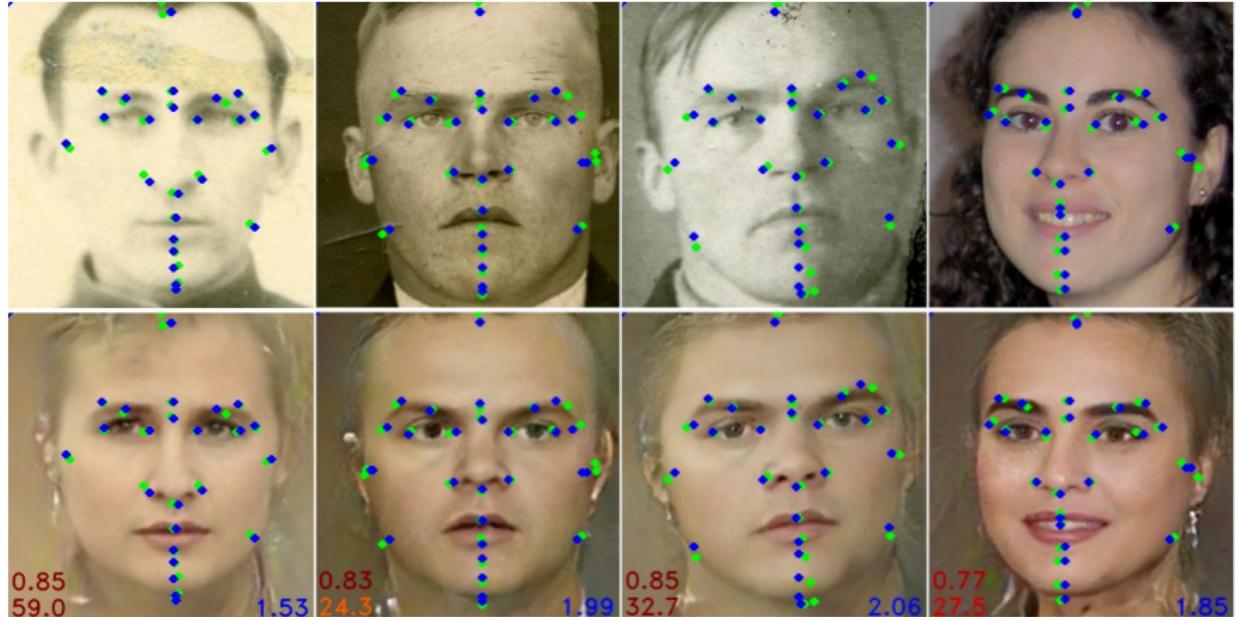


Media por landmark de las predicciones de **cross validation**:

Landmark	M. Base	M. Encoder	M. Decoder	M. Data Augmentation
Menton	4.176	3.997	3.991	3.006
Gnathion	2.629	2.606	2.634	1.497
Pogonion	2.509	2.546	2.585	1.359
Prosthion	1.178	1.048	0.97	0.815
Labiale Superius	1.965	2.04	2.059	1.531
Subnasale	2.056	1.992	2.089	1.272
Nasion	2.211	2.099	2.275	1.241
Glabella	2.562	2.736	2.912	1.545
Vertex	5.904	5.89	6.137	3.99
Left Gonion	5.652	5.789	5.665	1.846
Right Gonion	5.655	5.301	5.468	1.673
Left Zygion	5.351	5.288	5.29	3.322
Right Zygion	6.504	6.291	6.14	4.441
Left Alare	1.972	1.959	2.051	1.585
Right Alare	2.27	2.232	2.295	1.362
Left Endocanthion	1.675	1.71	1.827	1.227
Right Endocanthion	1.731	1.647	1.61	1.129
Left Exocanthion	1.7	1.733	1.809	1.186
Right Exocanthion	1.651	1.746	1.734	1.12
Left Tragion	4.15	3.633	4.163	2.066
Right Tragion	5.639	5.252	4.791	2.234
Labiale inferius	1.734	1.688	1.627	1.032
Trichion	4.445	4.762	5.196	3.37
Supramentale	1.933	1.991	1.839	1.109
Left Frontotemporal	2.37	2.452	2.474	1.386
Right Frontotemporal	2.97	3.282	3.009	1.826
Left Frontozygomaticus	1.566	1.499	1.525	1.036
Right Frontozygomaticus	2.483	2.516	2.395	1.057
Left Midsupraorbital	1.168	1.178	1.191	0.854
Right Midsupraorbital	1.429	1.348	1.48	0.945

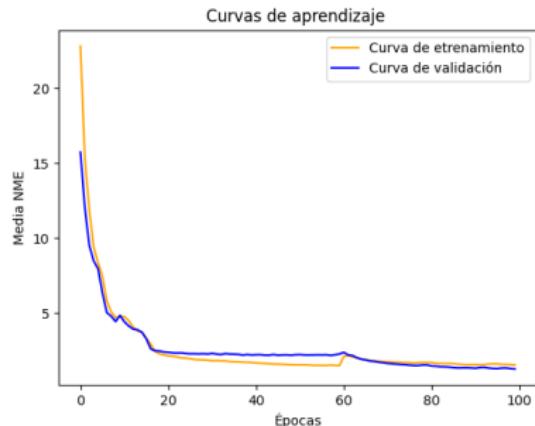
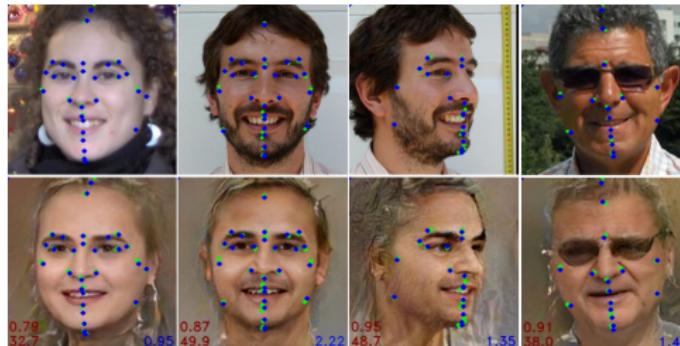


# Análisis Cualitativo: Modelo Data Augmentation



# Elección de modelo → Data Augmentation

- Entrenamos el modelo final con **todo** el conjunto de entrenamiento.
- No hay **overfitting**.



# Comparativa HF-ResNet<sup>16</sup>

## Diferencias:

- Métrica:

$$RMSE(y, \hat{y}) = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2}$$

Modelo	RMSE	P <sub>25</sub>	P <sub>50</sub>	P <sub>75</sub>
Modelo HyperFace-Resnet101	3.4106	1.802	2.7569	4.2849
Modelo 3FabRec	2.7666	1.9561	2.3812	3.1201

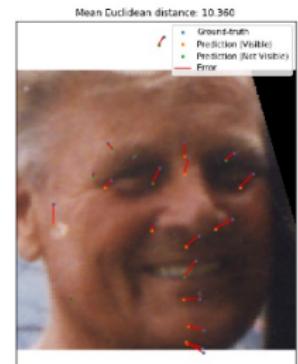
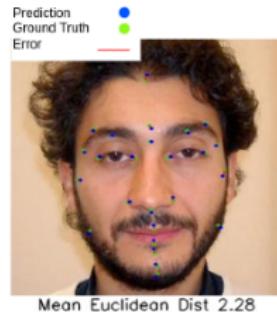
Landmark	RMSE 3FabRec	RMSE HyperFace-Resnet101
Menton	1.79	4.97
Gnathion	2.28	3.84
Pogonion	2.86	3.99
Prosthion	1.73	3.23
Labiale Superius	2.24	2.35
Subnasale	1.56	3.33
Nasion	1.51	3.12
Glabella	2.4	3.89
Vertex	4.3	8.83
Left Gonion	3.38	7.12
Right Gonion	2.3	6.11
Left Zygion	2.93	5.78
Right Zygion	3.23	6.94
Left Alare	1.17	3.50
Right Alare	1.42	2.84
Left Endocanthion	3.18	2.40
Right Endocanthion	4.56	2.32
Left Exocanthion	4.92	3.73
Right Exocanthion	6.98	3.62
Left Tragion	2.87	7.01
Right Tragion	2.65	6.41
Labiale inferius	2.02	3.11
Trichion	2.68	7.02
Supramentale	2.19	3.43
Left Frontotemporale	2.59	3.78
Right Frontotemporale	2.37	3.27
Left Frontozygomaticus	1.79	2.75
Right Frontozygomaticus	1.94	2.80
Left Midsupraorbital	1.44	2.57
Right Midsupraorbital	5.46	2.24

<sup>16</sup> G. G. Trenado (2019). "Aprendizaje Profundo y Antropología Forense". In: Universidad de Granada



# Comparativa HF-Resnet

- **Mejora** generalizada en el marcado.
- **Reducción** de la distancia euclídea media.
- **Mejora** el rendimiento en imágenes complicadas.



# Conclusiones

## Todos los objetivos cumplidos

- ① Investigación + comparación con el estado del arte.
- ② Estudio sobre los Autoencoders y redes adversarias.
- ③ Estudio de errores y preprocesamiento de los datos.
- ④ Estudio experimental.

Además, se han logrado otros **objetivos adicionales**:

- Entrenar 3FabRec con landmarks faltantes.
- Método robusto con few-shot learning.



# Preguntas



# Operador de dispersión

## Operador de dispersión

Sea  $\mathcal{P}_\infty$  el conjunto de todos los caminos finitos. La transformada de dispersión de  $f \in L^1(\mathbb{R}^d)$  se define para cualquier camino  $p \in \mathcal{P}_\infty$  como:

$$\overline{S}f(p) = \int_{\mathbb{R}^d} U[p]f(x)dx$$

Siendo  $\overline{S}f(p)$  invariante a traslaciones para un  $f$  fijo.

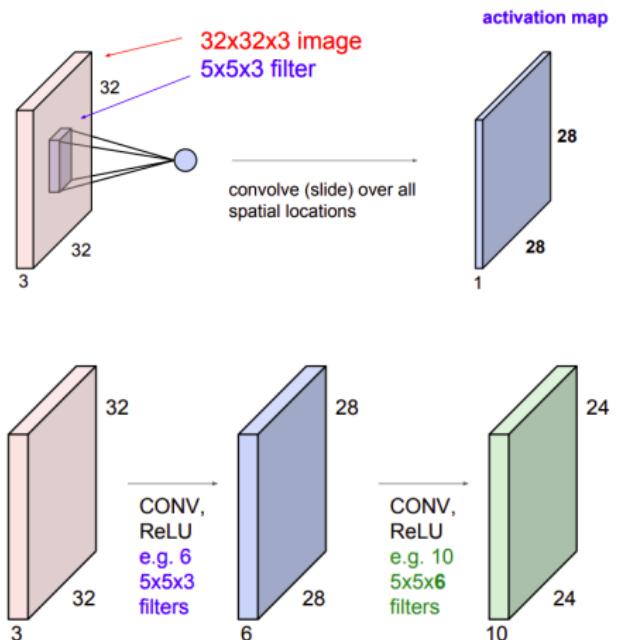


# Capas Convolucionales

## Características:

- Operación de convolución.
- Menor número de pesos.
- Mayor profundidad de red.

Imágenes extraída de <sup>17</sup>.



<sup>17</sup> Fei-Fei Li, Justin Johnson, Serena Young , Stanford University (2017). cs231n. = <http://cs231n.stanford.edu/2017/syllabus.html>,

# Capa de Pooling

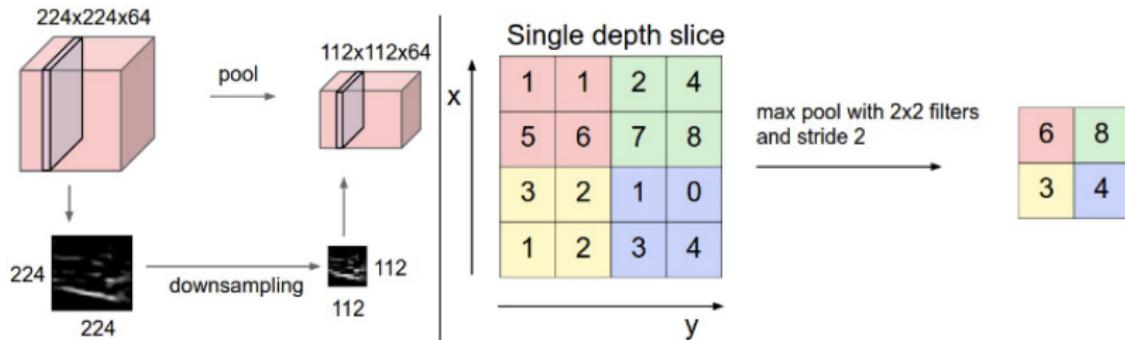
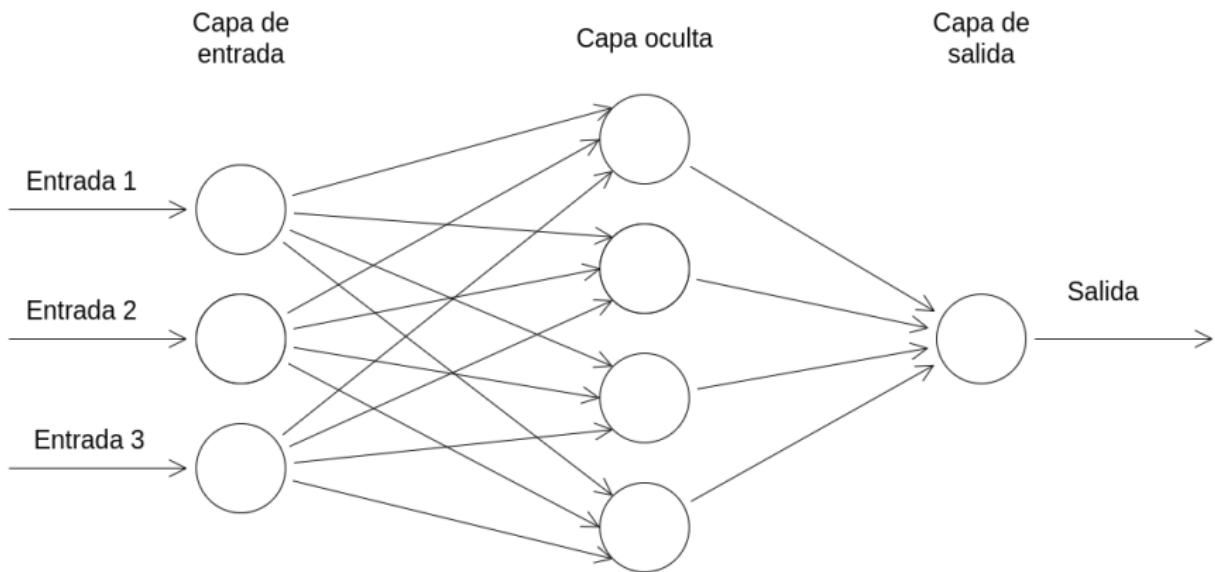


Figure: Imagen extraída de <sup>18</sup>.

<sup>18</sup>Fei-Fei Li, Justin Johnson, Serena Young , Stanford University (2017). cs231n. = <http://cs231n.stanford.edu/2017/syllabus.html>,



## Capa totalmente conectada



# Autoencoders

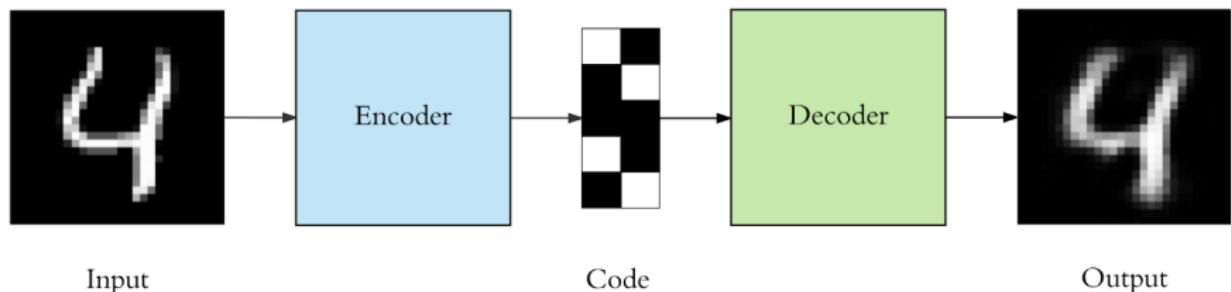
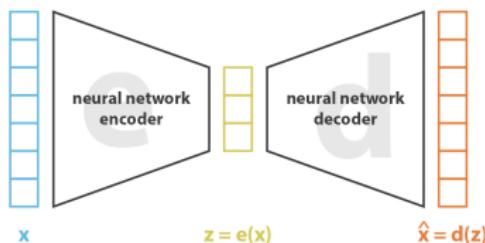


Figure: Imagen extraída de <sup>19</sup>.

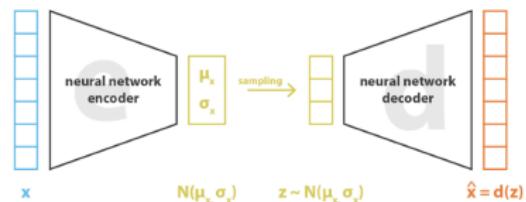
<sup>19</sup> A. Dertat (2017). *Applied Deep Learning.* =

<https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798>,

# Variational Autoencoder (VAE)



$$\text{loss} = \|x - \hat{x}\|^2 = \|x - d(z)\|^2 = \|x - d(e(x))\|^2$$



$$\text{loss} = \|x - \hat{x}\|^2 + \text{KL}[N(\mu_x, \sigma_x), N(0, I)] = \|x - d(z)\|^2 + \text{KL}[N(\mu_x, \sigma_x), N(0, I)]$$

<sup>20</sup> J. Rocca (2019b). *Understanding Variational Autoencoders (VAEs)*. = <https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73>,

# Generative Adversarial Network(GAN)

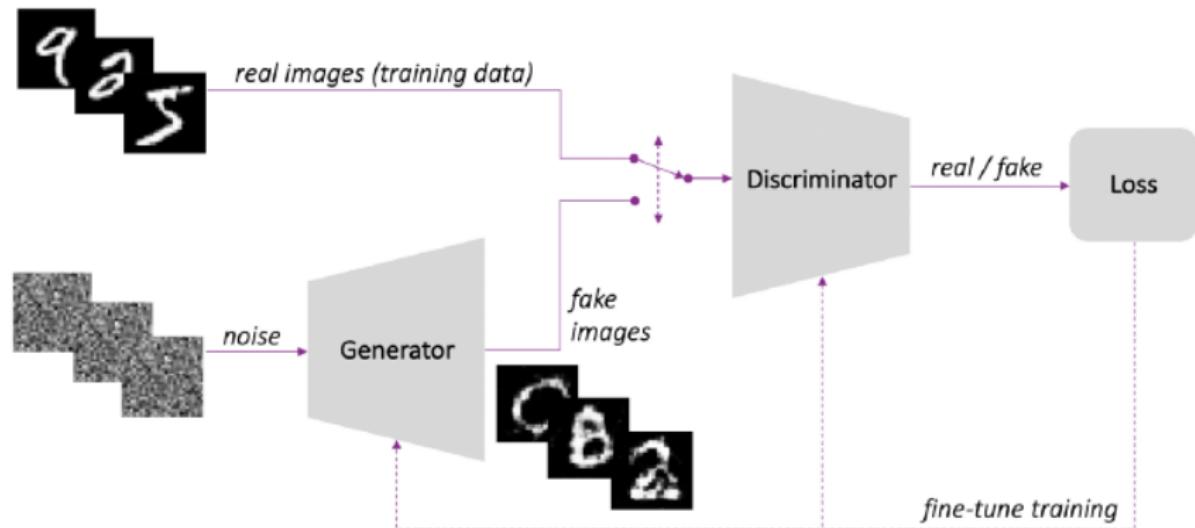


Figure: Imagen extraída de <sup>21</sup>.

<sup>21</sup> J. Rocca (2019a). *Understanding Generative Adversarial Networks (GANs)*. =  
<https://towardsdatascience.com/understanding-generative-adversarial-networks-gans-cd6e4651a29>,

# Taxonomía

- Encontramos 13 artículos en total.
- Sólo 4 relacionados con el marcado de landmarks cefalométricos.
- Emplean diversas técnicas de **visión por computador** y **aprendizaje automático**.

Autores	Año	Citas	Bancos de filtros	Dataset controlado	CNN	Landmarks
Mohd et al	2014	4	X	X		2
Galvánek et al	2015	18		X		14
Faria et al	2019	12	X	X	X	28
Anh Tuan et al	2022	1		X	X	16

# 3FabRec

## Función de pérdida parte no supervisada

$$\begin{aligned} \min_{E, G} \max_{D_z, D_x} \mathcal{L}_{AE}(E, G, D_z, D_x) = \\ \lambda_{rec} \mathcal{L}_{rec}(E, G) + \lambda_{cs} \mathcal{L}_{cs}(E, G) \\ + \lambda_{enc} \mathcal{L}_{enc}(E, D_z) + \lambda_{adv} \mathcal{L}_{adv}(E, G, D_x) \end{aligned}$$

## Función de pérdida parte supervisada

$$\mathcal{L}_H(ITL) = \mathbb{E}_{x \sim p(x)} [\|H - ITL(a)\|_2]$$

