Reconocimiento de expresiones faciales con redes profundas livianas usando Label Distribution Learning y el espacio de Action Units

Nicolás Mastropasqua[1], Daniel Acevedo[1,2]

[1] Departamento de Computación, Fac. de Cs. Exactas y Naturales, Universidad de Buenos Aires, Ciudad de Buenos Aires, Argentina. nmastropasqua@dc.uba.ar
[2] Instituto de Investigación en Cs. de la Computación (ICC). CONICET-UBA. Ciudad de Buenos Aires, Argentina. dacevedo@dc.uba.ar

Resumen En este trabajo nos enfocamos en el problema de Facial Expression Recognition (FER) y analizamos el uso de Label Distribution Learning en un modelo de Deep Learning liviano. Hoy en día, la búsqueda de soluciones 'lightweight' que logren resultados comparables a modelos de deep learning más robustos ha recibido particular atención debido a su implementación factible en dispositivos móviles. Además, considerando que la mayoría de los datasets de expresiones faciales suelen venir anotados con emociones categóricas cuando en realidad la mayoría de las expresiones exhibidas en escenarios 'in the wild' ocurren como combinaciones o composición de emociones básicas, hacemos uso de Label Distibution Learning (LDL) como estrategia para el entrenamiento. Asumimos también que las imágenes de expresiones faciales deberían tener una distribución de emoción similar a su vecindad en el espacio de etiquetas de Action Units. Esta información asociada a la distribución de los vecinos es capturada en la función de perdida para guiar el entrenamiento en LDL y así lograr mejorar los resultados de accuracy sobre el dataset RAFDB.

Keywords: Facial expression recognition · Label distribution learning · Ligthweight CNN.

1. Introducción

El reconocimiento de expresiones faciales (FER) está presente en numerosos campos de interés que presentan desafíos que van desde la asistencia de la conducción a través de la detección de fatiga según expresiones faciales [9] hasta la detección de depresión o clasificación del trastorno del espectro autista (ASD) [10] en el campo de la medicina.

Actualmente existen muchas arquitecturas de redes convolucionales que han ido progresando y consiguiendo muy buenos resultados en este tipo de problemas [3].Pero debido a la gran cantidad de capas empleadas, el costo en términos de FLOPs e incluso de memoria han ido intensificándose. En ocasiones, según la plataforma objetivo (drones, móviles, autos) puede que se tenga que restringir el

costo computacional para favorecer el uso de memoria o la latencia entre otros requerimientos. Con esta motivación se profundizó en el diseño de arquitecturas 'lightweight' que se ajustan al tradeoff de velocidad y accuracy. El problema es que las redes lightweight del estado del arte sufren cuando se las evalúa en escenarios de FER "in the wild", que muestran una gran variabilidad en las condiciones de iluminación, poses y oclusión.

Sin embargo los resultados obtenidos en Zhao et al. [16] sobre el dataset de RAFDB [11] 'in the wild' son alentadores ya que no solo logran un accuracy que mejora levemente con respecto a modelos del estado del arte de FER, si no que lo consiguen con la arquitectura lightweight 'EfficientFace' que tiene un orden de magnitud menos de cantidad de parámetros y MFLOPs. Para ello su arquitectura aprovecha el diseño de ShuffleNet v2 [13] como red backbone, agregando algunos bloques convolucionales para ser más robusta ante cambios en las poses y en oclusión pero sin dejar de lado la restricción de eficiencia. A su vez, para intentar compensar el sacrificio de complejidad de la red resultante proponen un método de entrenamiento más robusto usando Label Distribution Learning (LDL) [17] en lugar de un único label categórico, teniendo en cuenta que la mayoría de las expresiones faciales se corresponden con una combinación de emociones de distintas intensidades [14].

En este trabajo se propone en primer lugar replicar parte de los resultados reportados por Zhao et al. [16] donde se presenta la arquitectura EfficientFace junto a la técnica de entrenamiento LDL. En segundo lugar se busca mejorar la técnica de LDL aprovechando la topología del espacio de etiquetas de Action Units como en Chen et al. [2]. Estas anotaciones describen precisamente los movimientos de los músculos faciales según el Facial Action Coding System (FACS) [5] y pueden usarse para codificar emociones. De esta manera se espera que el modelo pueda beneficiarse de esta nueva combinación.

El resto del informe se estructura como sigue: En la Sec. 2 se desarrollan los métodos sobre los cuales se basa este trabajo y se discute su aplicación en este problema; en la Sec. 3 se muestran algunos de los resultados preliminares. Finalmente en la Sec. 4 se discuten las conclusiones del trabajo.

2. Métodos

Continuando con el trabajo propuesto por Zhao et al. [16], se utiliza el framework de EfficientFace (ver Fig. 1) para la tarea de reconocimiento de expresiones faciales (FER). De igual forma, para LDL se adopta una ResNet-50 [7] que se entrena como Generador de Distribuciones de Emociones (LDG, Label Distribution Generator) ante la ausencia de este tipo de anotaciones en RAFDB. A partir de esto, en nuestro trabajo se propone incorporar al entrenamiento del LDG el uso de la topología del espacio de etiquetas auxiliares de acuerdo a lo estudiado en Chen et al. [2] sobre RAFDB. En particular se elige la tarea de Action Unit Recognition (AUR) cuyas etiquetas deberían ser menos inconsistentes y estar sujetas a menor sesgo de anotación comparadas con las emociones. Además, al igual que en [2], se asume cierta noción de suavidad: imágenes que estén cercanas

en este espacio de etiquetas de AUs tienen mayor probabilidad de tener distribuciones de expresiones similares. Pensamos que esta estrategia debería asistir el entrenamiento del LDG favorablemente y en último lugar poder potenciar la performance de EfficientFace. La limitación de AUR es que la disponibilidad de datasets etiquetados no es demasiado alta, por lo tanto se anota el dataset con la herramienta del estado del arte OpenFace [1] que genera un vector de activación de un subconjunto de 18 AUs.

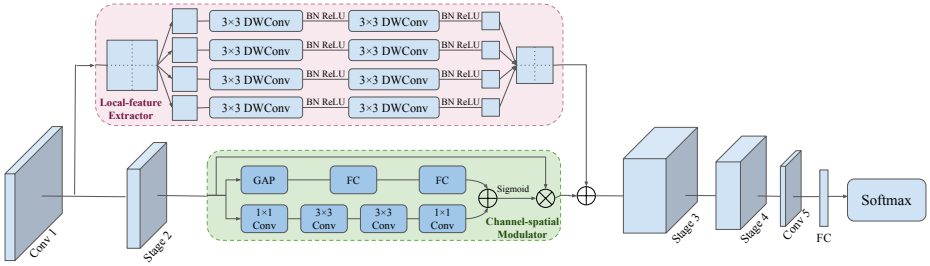

Figura 1. Esquema del modelo EfficientFace [16].

2.1. Label Distribution Learning

Según [14], solo existen una cantidad reducida de emociones básicas y el resto ocurren como combinaciones de éstas con cierta intensidad. Por esta razón, utilizar una única etiqueta de emoción (SLL, Single Label Learning) puede incorporar un grado no menor de ambigüedad ya que no permite capturar toda esta expresividad, más aún en un contexto in the wild. Además, este tipo de etiquetas suele estar sujeta a sesgos de anotación dado por el origen y contexto cultural de los anotadores. Si consideramos a cada emoción básica como una etiqueta, Multi Label Learning permite describir a cada imagen a partir de un conjunto de etiquetas definidas como relevantes. El problema es que se pierde noción de la intensidad con la que aparece cada una.

Utilizando la distribución de emociones [17] se asocia a cada imagen un vector como etiqueta, donde cada componente es la intensidad que aporta una determinada emoción básica a la expresión facial subyacente. Normalizando cada intensidad al rango $[0, 1]$ y haciendo que la suma de las componentes de 1, se obtiene la denominada distribución de la emoción para la imagen dada. LDL debería, entonces, ser más robusto para la tarea en cuestión al tratar de mitigar los problemas mencionados. Si bien existen algunos datasets usados en FER que anotan esta distribución [12], no es lo más común y por lo tanto, como en [16], se entrena una Resnet-50 con el objetivo de aprender a generar distribuciones de emociones (LDG). Es importante notar que este componente (Resnet-50) se entrena con SLL y luego se lo utiliza congelado en el entrenamiento de EfficientFace como generador de ground-truth. Es decir, durante el entrenamiento

de la EfficientFace, la salida esperada de la red con la cual se computa el error de aprendizaje será la provista por la Resnet-50 que está entrenada como un generador de distribuciones.

Bajo el supuesto de suavidad, se define una función de pérdida auxiliar $\Omega(\theta)$ (ver Eq. 1) como parte de la función de pérdida del LDG-AUs para que dada una instancia se tenga en cuenta la predicción de la distribución de emociones de los vecinos según el espacio de etiquetas de AUs. Cada predicción del vecindario estará pesada por una noción de similaridad local como en [2].

2.2. El framework de EfficientFace resultante

La arquitectura de la red backbone de EfficientFace se basa en ShuffleNet v2 [13]. Para adaptarse a escenarios in the wild, agrega en sus etapas iniciales un Local-Feature Extractor que aprende features de regiones locales partiendo la imagen en cuatro parches y empleando una serie de 'depthwise-convolutions' [4] para reducir el costo computacional. A su vez incorpora un Channel-Spatial Modulator que busca resaltar features faciales globales relevantes que resultan de las primeras etapas de extracción de features. Finalmente se computan local-global features que serán usados en las sucesivas etapas combinando el resultado de estos dos bloques propuestos (ver Fig. 1).

La función de pérdida utilizada para entrenar es Cross-Entropy, donde la salida de la red para una instancia dada será un vector de distribución de emociones y el ground-truth asociado será el vector de distribución de emociones estimado por el LDG para esa misma instancia. Al momento de hacer inferencia, se toma el índice del máximo en el vector de probabilidades como la etiqueta predicha de la emoción.

Por otro lado, para entrenar el LDG-AUs empleando la idea propuesta en las secciones previas, se genera en primer lugar un grafo aKNN [8] para todo el dataset según las distancia euclidiana entre los vectores de AUs (l_i). Luego, se tiene para cada índice de instancia una lista con los índices de sus k vecinos junto con el coeficiente de similaridad local $a_{ij} = \exp\left(-\frac{\|l_i - l_j\|}{\sigma^2}\right)$ como en [2].

Siendo $f(x_i|\theta)$ el vector softmax de salida de la red con parámetros θ para la entrada x_i (distribución de emoción para x_i) y $D_{KL}(P, Q)$ la Divergencia de Kullback-Leibler entre las distribuciones P y Q, se computa la función de pérdida auxiliar como:

$$\Omega(f(x|\theta)) = \sum_{ij} a_{ij} D_{KL}(f(x_j)||f(x_i)) \tag{1}$$

A diferencia de [3], que utiliza Divergencia de Kullback-Leibler entre la salida del generador y el ground-truth, se define $L(\theta)$ la función de pérdida Cross-Entropy ya que rápidamente muestra señales de entrenamiento. Finalmente, la función de perdida utilizada para el LDG-AUs es $L(\theta)_{ldg} = L(\theta) + \lambda\Omega(\theta)$.

3. Experimentos y Resultados

Se utiliza la parte del dataset RAFDB [11] que contiene imágenes anotadas con las siete emociones básicas y cuenta con 12271 imágenes para train y 3068

para test. En todos los casos las redes fueron pre-entrenadas con el dataset de reconocimiento facial MS-Celeb-1M [6]. Se usan todas las imágenes ya alineadas y se las escala a 224×224. También se aplican las transformaciones random horizontal flipping y random cropping. Esta última demostró evitar que la red tienda a hacer overfitting.

Se hicieron exhaustivas pruebas para determinar el mejor entrenamiento para cada modelo. Para verificar la efectividad del LDG sobre EfficientFace se hizo un pequeño estudio de ablación. A su vez se tiene en cuenta, como referencia, el desempeño del LDG usado como clasificador. Finalmente, se compara el accuracy entre nuestro modelo EF+LDG-AUs y el replicado EG+LDG. EF SLL fue entrenada durante 100 epochs con un optimizador SGD con bs (batch-size) de 128, momentum 0.9, weight decay 0.0004, lr (learning rate) inicial 0.01 y decaimiento exponencial cada 15 epochs. LDG fue entrenado con la misma configuración anterior, pero con bs de 16 y lr inicial 0.001. LDG-AUs fue entrenado con la misma configuración de LDG, pero durante 35 epochs por restricciones de tiempo. A su vez se utilizaron 4 vecinos para armar el grafo y λ de $L(\theta)_{ldg}$ en 0.01. Ambos modelos de EfficientFace con LDG fueron entrenados durante 100 epochs usando SGD con bs 256 y One-Cycle Scheduler [15] con lr máximo de 0.7. Para cada modelo se repite el entrenamiento cinco veces con distintas seeds. Todos los experimentos fueron implementados con Pytorch entrenados en un servidor con dos GeForce GTX 1080 Ti.

Cuadro 1. Accuracy de los métodos estudiados sobre el test set de RAFDB.

	EF + LDG-AUs	EF + LDG	EF SLL	LDG-AUs	LDG
Accuracy (max)	88.29	88.10	86.47	88.23	88.36
(med)	87.87	88.16	86.47	87.77	87.84
(min)	87.80	87.87	85.98	87.12	86.86

La tabla 1 muestra los resultados de nuestro trabajo. En primer lugar el experimento de ablación se encuentra en concordancia con lo reportado en [16]. Se observa una mejora de 1,63 % al comparar EF SLL con EF+LDG, mostrando la ventaja de utilizar LDL. Diferente de [16], al pensar al LDG como clasificador, se logra un accuracy superior al entrenar el LDG (88.36) contra 86.93 reportado allí. De esta forma EF+LDG no supera con claridad a este modelo del estado del arte pero lo notable es que alcanza un desempeño comparable con un diseño lightweight. Finalmente encontramos que en los resultados preliminares de nuestra versión LDG-AUs, evaluado como clasificador o bien como parte de EF, son muy similares a sus versiones base.

4. Conclusiones

En el trabajo pudimos confirmar los resultados de eficiencia de LDL empleados en EfficientFace sobre el dataset RAFDB en [16]. Los resultados preliminares

sobre nuestro modelo LDG-AUs sugieren que se podría continuar explorando esa dirección. Las anotaciones obtenidas de AUs podrían no ser precisas para el contexto in the wild, terminando por generar ruido que dificulte el entrenamiento. Por eso se piensa en aplicar alguna técnica de detección de anomalías. A futuro también será de interés evaluar la generalización cross-dataset del framework y la tolerancia a inconsistencia de anotaciones de emociones.

Referencias

1. Baltrušaitis, T., Mahmoud, M., Robinson, P.: Cross-dataset learning and person-specific normalisation for automatic action unit detection. In: 11th IEEE Int. Conf. and Workshops on Aut. Face and Gesture Recognition (FG). vol. 06, pp. 1–6 (2015)
2. Chen, S., Wang, J., Chen, Y., Shi, Z., Geng, X., Rui, Y.: Label distribution learning on auxiliary label space graphs for facial expression recognition. In: IEEE/CVF Conf. on Computer Vision and Pattern Recog. (CVPR). pp. 13981–13990 (2020)
3. Chen, W., Zhang, D., Li, M., Lee, D.J.: Stcam: Spatial-temporal and channel attention module for dynamic facial expression recognition. IEEE Trans. on Affective Computing (2020)
4. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Conf. on Comp. Vision and Patt. Recog. (CVPR). pp. 1800–1807. USA (2017)
5. Ekman, P., Friesen, W.V.: Facial action coding system: a technique for the measurement of facial movement (1978)
6. Guo, Y., Zhang, L., Hu, Y., He, X., Gao, J.: Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In: Computer Vision – ECCV. pp. 87–102 (2016)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conf. on Comp. Vision and Pattern Recog. (CVPR). pp. 770–778 (2016)
8. Iwasaki, M., Miyazaki, D.: Optimization of indexing based on k-nearest neighbor graph for proximity search in high-dim. data. ArXiv **abs/1810.07355** (2018)
9. Khan, S.A., Hussain, S., Xiaoming, S., Yang, S.: An effective framework for driver fatigue recognition based on intelligent facial expressions analysis. IEEE Access **6**, 67459–67468 (2018)
10. Li, B., Mehta, S., Aneja, D., Foster, C., Ventola, P., Shic, F., Shapiro, L.: A facial affect analysis system for autism spectrum disorder. In: IEEE Int. Conf. on Image Processing (ICIP). pp. 4549–4553 (2019)
11. Li, S., Deng, W., Du, J.: Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In: IEEE/CVF Conf. on Computer Vision and Pattern Recog. (CVPR). pp. 2584–2593 (2017)
12. Lyons, M., Kamachi, M., Gyoba, J.: Japanese female facial expression (jaffe) database (7 2017)
13. Ma, N., Zhang, X., Zheng, H.T., Sun, J.: Shufflenet v2: Practical guidelines for efficient cnn arch. design. In: Proc. of Eur. Conf. on Comp. Vision (ECCV) (2018)
14. Plutchik, R.: Chapter 1. a general psychoevolutionary theory of emotion. In: Plutchik, R., Kellerman, H. (eds.) Theories of Emotion, pp. 3–33. Ac. Press (1980)
15. Smith, L.N., Topin, N.: Super-convergence: very fast training of neural networks using large learning rates. In: Proc. of SPIE. vol. 11006, pp. 369–386 (2019)
16. Zhao, Z., Liu, Q., Zhou, F.: Robust lightweight facial expression recognition network with label distribution training. Proc. of the AAAI Conf. on Artificial Intelligence **35**(4), 3510–3519 (2021)
17. Zhou, Y., Xue, H., Geng, X.: Emotion distribution recognition from facial expressions. p. 1247–1250. Assoc. for Computing Machinery, NY, USA (2015)