

ANÁLISIS DE LOS EFECTOS DE LA INCLUSIÓN DE UN STN EN NUESTRA RED

Por Antonio Jesús García Nieto

CONTROL DE VERSIONES

[illegible]

TABLA DE CONTENIDO

Control de versiones.....	2
Introducción	4
Planteamiento del estudio	4
Resultados	6
Los resultados de la prueba 1.....	6
Los resultados de la prueba 2.....	6
Los resultados de la prueba 3.....	6
Conclusión del experimento.....	7
Bibliografía.....	7

INTRODUCCIÓN

En este experimento estudiaremos la variación de la precisión de nuestra red en función de la introducción de un nuevo módulo denominado STN. Este módulo apareció en (Jaderberg et al., 2015). Este pretende dar solución o al menos servir de alternativa a las capas de max pooling.

Este módulo se puede integrar perfectamente en la red, y no necesita una clasificación previa de los datos de entradas en la red.

En este estudio se pretende crear un sistema que aplique una transformación espacial a la imagen de entrada, de forma que amplíe y se focalice la zona afectada por la lesión.

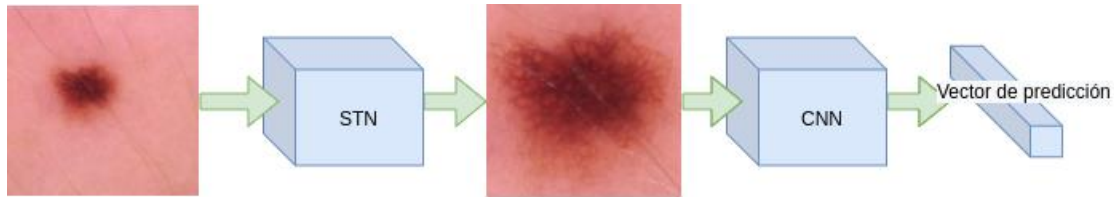


Ilustración 1. Sistema formada por STN y clasificador

La hipótesis que pretendemos validar en este estudio es:

1. *Incluir un STN en la primera capa de la red podría incrementar el rendimiento del clasificador.*

PLANTEAMIENTO DEL ESTUDIO

El estudio consistirá en una comparación de la precisión media entre distintas configuraciones del modelo.

La configuración general será:

En lugar de usar la generación típica de imágenes (data augmentation), para este experimento se trabajará con 7000 imágenes, con un tamaño de 128x128 en lugar de 50x50. El STN generará un recorte de 50x50 de cada imagen, que será el tamaño con el que trabaje el clasificador.

Se seguirá el mismo esquema de entrenamiento que en el experimento 1. Recordemos que el este es:

- El conjunto de entrenamiento = source (60% del total del conjunto de datos),
- El conjunto de reentrenamiento = train (70% del 40% restante del conjunto de datos),
- El conjunto de test = test (21% del 40% restante del conjunto de datos)
- El conjunto de validación = validación (9% del 40% restante del conjunto de datos)

Los hiperparámetros serán los siguientes:

- Ratio de aprendizaje de 0.001 y optimizador Adam.
- Épocas: 25 para entrenamiento y para transferencia.
- Callbacks: Se detiene el entrenamiento si la red consigue peores resultados en 3 ocasiones consecutivas.
- Tamaño de lote: 16

Se realizarán tres pruebas:

1. **Prueba 1.** Se entrenará en primer lugar el clasificador (según el esquema seguido hasta ahora, aplicando transfer learning), y posteriormente se congelará completamente para pasar a entrenar la STN.
2. **Prueba 2.** Se entrenará en primer lugar el clasificador (según el esquema seguido hasta ahora, aplicando transfer learning), y posteriormente se reentrenará con la STN.
3. **Prueba 3.** Se entrenará en primer lugar el clasificador (según el esquema seguido hasta ahora, aplicando transfer learning), y posteriormente se reentrenará con la STN, en una versión con mayor tamaño.

Para asegurar la robustez de la prueba se entrenará 10 veces cada modelo con cada configuración.

A continuación, se muestra la configuración del clasificador:

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 50, 50, 3)]	0
conv2d (Conv2D)	(None, 48, 48, 32)	896
batch_normalization (Batch Normalization)	(None, 48, 48, 32)	128
re_lu (ReLU)	(None, 48, 48, 32)	0
conv2d_1 (Conv2D)	(None, 46, 46, 32)	9248
batch_normalization_1 (Batch Normalization)	(None, 46, 46, 32)	128
re_lu_1 (ReLU)	(None, 46, 46, 32)	0
batch_normalization_2 (Batch Normalization)	(None, 46, 46, 32)	128
conv2d_2 (Conv2D)	(None, 44, 44, 64)	18496
batch_normalization_3 (Batch Normalization)	(None, 44, 44, 64)	256
re_lu_2 (ReLU)	(None, 44, 44, 64)	0
max_pooling2d (MaxPooling2D)	(None, 22, 22, 64)	0
flatten (Flatten)	(None, 30976)	0
dense (Dense)	(None, 128)	3965056
p_re_lu (PReLU)	(None, 128)	128
dense_1 (Dense)	(None, 7)	903
Total params: 3,995,367		
Trainable params: 3,995,047		
Non-trainable params: 320		

En las siguientes imágenes se muestra las configuraciones de la red de localización utilizada en la prueba 1 y 2, y la versión extendida usada en la prueba 3

La red de localización del STN seguirá una estructura muy simple.

1. Capa de convolución con 32 filtros 3x3, y función de activación ReLU
2. Capa de Max Pool 2x2.
3. Capa de convolución con 64 filtros 3x3, y función de activación ReLU
4. Capa de Max Pool 2x2.
5. Capa de Flatten
6. Capa Dense con 32 unidades y función de activación ReLU
7. Capa de Dropout con coeficiente de 0.5
8. Capa Dense con 6 unidades que son los parámetros que conformaron theta

La versión ampliada usará el doble de filtros, es decir:

1. Capa de convolución con 64 filtros 3x3, y función de activación ReLU
2. Capa de Max Pool 2x2.
3. Capa de convolución con 128 filtros 3x3, y función de activación ReLU
4. Capa de Max Pool 2x2.
5. Capa de Flatten
6. Capa Dense con 32 unidades y función de activación ReLU
7. Capa de Dropout con coeficiente de 0.5
8. Capa Dense con 6 unidades que son los parámetros que conformaron theta

RESULTADOS

LOS RESULTADOS DE LA PRUEBA 1.

	CNN	CNN_STN					
0	0.816193	0.818381	10	0.810357	0.811087		
1	0.830780	0.830780	11	0.781182	0.777535		
2	0.820569	0.820569	12	0.824216	0.823487		
3	0.840263	0.840992	13	0.757841	0.757112		
4	0.830780	0.834427	14	0.800146	0.799416		
5	0.777535	0.779723	15	0.820569	0.825675		
6	0.825675	0.826404	16	0.766594	0.766594		
7	0.814004	0.813275	17	0.731583	0.727936	CNN	0.802881
8	0.780452	0.781911	18	0.819840	0.819110	CNN_STN	0.803428
9	0.794311	0.798687	19	0.814734	0.815463		

LOS RESULTADOS DE LA PRUEBA 2.

	CNN	STN_CNN		
0	0.816193	0.818381		
1	0.830780	0.830780		
2	0.820569	0.820569		
3	0.840263	0.840992		
4	0.830780	0.834427		
5	0.777535	0.779723		
6	0.825675	0.826404		
7	0.814004	0.813275	CNN	0.813056
8	0.780452	0.781911	STN_CNN	0.814515
9	0.794311	0.798687		

LOS RESULTADOS DE LA PRUEBA 3

	CNN	STN_CNN		
0	0.822757	0.817651		
1	0.824216	0.825675		
2	0.814004	0.814004		
3	0.822757	0.825675		
4	0.819110	0.816193		
5	0.830780	0.827133		
6	0.822757	0.824945		
7	0.750547	0.755653	CNN	0.815171
8	0.835886	0.832239	STN_CNN	0.815244
9	0.808899	0.813275		

CONCLUSIÓN DEL EXPERIMENTO

Ante los resultados obtenidos podemos concluir que incluir STN en las capas superiores de un clasificador es una técnica que no aumenta en gran medida la precisión, tal y como se describe en (Jaderberg et al., 2015) , donde la mejora sobre MNIST es de apenas 0.3%. No obstante, tampoco empeora los resultados de esta.

Uno de los motivos que quizás influya en nuestros resultados es que las imágenes de nuestro conjuntos de datos apenas tienen características estandarizables, por un lado, la clasificación de las lesiones de piel no depende de la rotación, y por otro suelen estar bastantes bien centradas.

A nivel computacional, incluir este modulo no ha supuesto una carga relevante. Lo que puede incentivar la inclusión de esta.

BIBLIOGRAFÍA

Jaderberg, M., Simonyan, K., Zisserman, A., & Kavukcuoglu, K. (2015). Spatial Transformer Networks. *Advances in Neural Information Processing Systems, 2015-January*, 2017–2025.
<http://arxiv.org/abs/1506.02025>