Using preprocessing as a tool in medical image detection

Mathias Kirkerød [1,3], Vajira Thambawita[1,2], Michael Riegler[1,2,3], Pål Halvorsen[1,3]
[1]Simula Research Laboratory, Norway
[2]Oslo Metropolitan University
[3]University of Oslo
mathias.kirkerod@gmail.com,vajira@simula.no,michael@simula.no,paalh@simula.no

ABSTRACT

In this paper we describe our approach to gastrointestinal disease classification for the medico task at MediaEval 2018. We propose multiple ways to inpaint problematic areas in the test and training set to help with classification. We discuss the effect that preprocessing does to the input data with respect to removing regions with sparse information. We also discuss how preprocessing affects the training and evaluation of a dataset that is limited in size. We will also compare the different inpainting methods with transfer learning using a convolutional neural network.

1 INTRODUCTION

Medical image diagnosis is a challenging task in the industry of computer vision. In the last couple of years, as computing power has increased, machine learning has become a tool in the task of image detection, segmentation and classification. In this paper we are looking in depth how to use machine learning to help solve classification tasks on the data-set from the Medico task [8]. The Medico task focuses on image classification in the gastrointestinal (GI) tract. The data is divided in to 16 different classes.

Similar to other parts of image detection, the Medico dataset encounter the challenges that the amount of data is too small, or that the training data does not cover the full distribution of the data in the test case. The main goal of this task is to classify medical images. Our proposal is to use unsupervised machine learning for removal of the green corners that are in the Medico dataset. The details of the task are described in [5, 7].

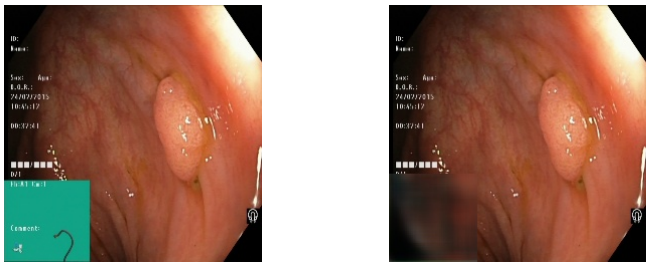
2 APPROACH

Our approach is divided in to two steps: first preprocessing, then classifying. Our focus is mainly on the preprocessing of the data to remove the green corners in the medical images.

After the preprocessing the dataset we run it through a Convolutional Neural Network (CNN) based on transfer learning. We chose the CNN model based on the top 5 and top 1 accuracy of the pre-trained networks on the Keras documentation pages.

In our approach we use the InceptionResNetV2 [9] network. We also remove the top layer and replace it with a global average pooling layer and a dense 16 layer output, to match the number of classes wanted. In addition, we do not freeze any layers of the model. The five submissions that we run is with the same hyperparameters in the transferlearning model. This means that the difference in results should only come from the different training datasets we use.

MediaEval'18, 29-31 October 2018, Sophia Antipolis, France



(a) Image before inpainting **(b) Image after inpainting**

Figure 1: Differences of images after inpainting

The medical data has 1 main feature that we focus on during the preprocessing, namely the green square in the bottom left corner. A neural network often struggle with areas with really sparse information. Our hypothesis is that just replacing the green area with a similar black area will not yield a better result.

We have a dataset that we use as a base case. This dataset was not augmented, other than shrinking the size of every image to a fixed resolution. The other datasets were augmented in a way that would cover up the green square in one way or another.

Our hypothesis it that if we recreate the areas as they would look like without any sparse areas, the classifier can focus on the right features for classifications. We propose 4 different methods on how to inpaint the corner area of the medical images.
An autoencoder [4], a context conditional generative adversarial network[2, 3], a context encoder [6], and a simple crop of the image.

2.1 Autoencoder

For the autoencoder approach, we created and trained a custom autoencoder from scratch. Our autoencoder consist of a encoder-decoder network, with 2D convolutions as well as rectified linear units as activation functions. In the layer between the encoder and the decoder we included a 25% dropout. [1]

To preprocess the medical data we feed the whole image through the encoder-decoder network. We take the loss of the whole reconstructed image, but only keep the inpainted part. Under training, the goal is to minimize the loss: $L(x, g(f(\tilde{x})))$ Where x is an image without a green corner, and $\tilde{x}$ is the same image with an artificial green corner. In theory we can replace any part of the image with this method.

Table 1: Validation set' results

Method	REC	PREC	SPEC	ACC	MCC	F1
Autoencoder	0.929	0.929	0.981	0.929	0.923	0.928
CC-GAN	0.931	0.932	1.000	**0.931**	**0.926**	0.931
Contextencoder	0.926	0.928	0.945	0.926	0.920	0.926
Clipping	0.903	0.904	0.980	0.903	0.895	0.903
Non-augmenteted	0.925	0.927	0.981	0.925	0.919	0.924

2.2 Context encoder

For the context encoder approach, we created a new encoder-decoder network. Here the encoder has a similar structure to the autoencoder, but our decoder is only making outputs at the size of the desired area to inpaint. In addition to the loss generated from taking a MSE loss[6]:
$L(\hat{x}, g(f(x)))$ *Where* $\hat{x}$ *is an image with an artificial green corner, and* x *is the part that was replaced by the corner,* we include an adversarial loss, as described in [6].

With the context encoder we feed images without a green corner in to the encoder-decoder network. The output of the network is the same size as the area we want to fill.

2.3 Context conditional generative adversarial network

For the generative adversarial approach, we create a similar structure as the autoencoder. We have a constant 10% dropout at each layer in the discriminator. As with the autoencoder we have the same size input as output, but we only decide to keep the parts we want to inpaint.

We use the same type of loss as the context encoder, with 15% of the loss coming from a MSE loss, and the remaining 85% coming from the adversarial loss.

2.4 Clipping instead of inpainting

The last method was just to crop the images in a way that excluded the green corner. Since every image is scaled down to 256x256 px during preprocessing, the same is done with the clipped version (after the clip the size was reduced to 256x256).

The clipping was done in a way so that we had the most amount of center frame, and minimal amount of the bottom left corner, without sacrificing to much of the image.

3 RESULTS AND ANALYSIS

We made the augmented datasets before we trained the preprocessing model. This means that the transferlearning model did not augment the images at runtime. We split the data into a 70% train set, and a 30% validation set.

Our results on the test set are tabulated in Table 1. The official Results on the test set are tabulated in Table 2. Table 3 shows the confusion matrix from the CC-GAN from the official test set.

The results show that the CC-GAN got the highest MCC score with 0.926, and also the most realistic inpaintings. The context encoder had the lowest MCC score with 0.920, and also the worst inpainted areas. The official result did have the same pattern in MCC score, though the base case got the best result. In both cases the clipping gave significantly worse result.

As expected, most of the images was classified correctly, but we had some problems distinguishing between esophagitis and normal-z-line. We also had a few cases of instruments where there were none.

Table 2: Official Results

Method	REC	PREC	SPEC	ACC	MCC	F1
Autoencoder	0.915	0.915	0.994	**0.989**	0.910	0.915
CC-GAN	0.915	0.915	0.994	**0.989**	0.910	0.915
Contextencoder	0.910	0.910	0.994	0.988	0.905	0.910
Clipping	0.904	0.904	0.993	0.988	0.898	0.904
Non-augmenteted	0.917	0.917	0.994	**0.989**	**0.911**	0.917

Table 3: Confusion Matrix

A:ulcerative-colitis , **B**:esophagitis , **C**:normal-z-line , **D**:dyed-lifted-polyps , **E**:dyed-resection-margins , **F**:out-of-patient , **G**:normal-pylorus , **H**:stool-inclusions , **I**:stool-plenty , **J**:blurry-nothing , **K**:polyps , **L**:normal-cecum , **M**:colon-clear , **N**:retroflex-rectum , **O**:retroflex-stomach , **P**:instruments

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
A	510	0	1	0	1	0	1	0	69	0	5	24	0	3	0	13
B	3	401	68	0	1	0	5	0	0	0	0	0	0	0	1	0
C	0	153	489	0	0	0	3	0	0	0	0	0	0	0	0	0
D	0	0	0	502	39	0	0	0	0	0	3	0	0	1	0	45
E	0	0	0	46	517	1	0	0	0	0	1	0	0	0	0	15
F	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G	2	2	3	0	0	0	547	0	0	0	0	0	0	0	1	0
H	0	0	0	0	0	0	0	486	35	0	0	0	0	0	0	0
I	3	0	0	0	2	0	0	1	1857	0	3	1	0	0	0	3
J	1	0	0	0	0	1	0	1	0	36	0	0	1	0	0	0
K	8	0	1	5	2	3	4	0	0	0	349	17	0	2	1	55
L	11	0	1	2	1	0	1	0	1	1	11	542	0	0	0	3
M	2	0	0	0	0	0	0	18	2	0	1	0	1064	0	1	3
N	2	0	0	1	1	0	0	0	0	0	1	0	0	183	4	5
O	0	0	0	0	0	0	0	0	1	0	0	0	0	2	389	0
P	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	131

4 CONCLUSION

In general, when training on a dataset that is homogeneous, the preprocessing is less valuable. We want to remove areas with sparseness, and areas that has nothing to do with the classification.
In our example we used 3 different methods to do this, and we had no improvements in the results. As we can see from the validation set, we saved under a percent on the best method, and we got a worse score on the official results.
We conclude that preprocessing the Medico dataset is not worth the hassle. The effort put in to preprocess the images yields little to no improvement to the result. We recommend that the time is used to find the right network, with the right hyper-parameters instead. A reason to lackluster results might be caused that the training and the test set have the same green squares in the same classes. We suspect that the similarity in the test and train set makes the squares an essential part of the image. We believe that the result would be much better if the test set would be completely without the squares, as they would if they were "real time" images.

In a future test we would also recommend removing the four black edges too. With the images being round, this might be a challenge, since there are no full-resolution images (without zoom) that captures the edges. With the medico dataset, this method will probably not give a better score, on the basis that every image in the dataset has the same four black corners.

REFERENCES

[1] Aaron Courville Yoshua Bengio David Warde-Farley, Ian J. Goodfellow. 2013. An empirical analysis of dropout in piecewise linear networks. abs/1609.05158 (2013). arXiv:1312.6197v2 https://arxiv.org/pdf/1312.6197v2

[2] Emily L. Denton, Sam Gross, and Rob Fergus. 2016. Semi-Supervised Learning with Context-Conditional Generative Adversarial Networks. *CoRR* abs/1611.06430 (2016). arXiv:1611.06430 http://arxiv.org/abs/1611.06430

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.

[4] Y. Kamp H. Bourlard. 1988. Auto-Association by Multilayer Perceptrons and Singular Value Decomposition. (1988). http://ace.cs.ohio.edu/~razvan/courses/dl6890/papers/bourlard-kamp88.pdf

[5] Pål Halvorsen Thomas de Lange Kristin Ranheim Randel Duc-Tien Dang-Nguyen Mathias Lux Konstantin Pogorelov, Michael Riegler. 2018. Mediaeval information. http://multimediaeval.org/mediaeval2018/medico/. (2018). Accessed: 2018-10-16.

[6] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. 2016. Context Encoders: Feature Learning by Inpainting. *CoRR* abs/1604.07379 (2016). arXiv:1604.07379 http://arxiv.org/abs/1604.07379

[7] Konstantin Pogorelov, Kristin Ranheim Randel, Carsten Griwodz, Sigrun Losada Eskeland, Thomas de Lange, Dag Johansen, Concetto Spampinato, Duc-Tien Dang-Nguyen, Mathias Lux, Peter Thelin Schmidt, Michael Riegler, and Pål Halvorsen. 2017. KVASIR: A Multi-Class Image Dataset for Computer Aided Gastrointestinal Disease Detection. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. ACM, New York, NY, USA, 164–169. https://doi.org/10.1145/3083187.3083212

[8] Konstantin Pogorelov, Michael Riegler, Pål Halvorsen, Thomas De Lange, Kristin Ranheim Randel, Duc-Tien Dang-Nguyen, Mathias Lux, and Olga Ostroukhova. 2018. Medico Multimedia Task at MediaEval 2018. (2018).

[9] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. 2016. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *CoRR* abs/1602.07261 (2016). arXiv:1602.07261 http://arxiv.org/abs/1602.07261