Capsule Networks against Medical Imaging Data Challenges

Amelia Jiménez-Sánchez, Shadi Albarqouni, Diana Mateus

Capsule Networks against Medical Imaging Data Challenges

Amelia Jiménez-Sánchez[1], Shadi Albarqouni[2], Diana Mateus[3]

[1] BCN MedTech, DTIC, Universitat Pompeu Fabra, Spain.
[2] Computer Aided Medical Procedures, Technische Universität München, Germany.
[3] Laboratoire des Sciences du Numérique de Nantes, UMR 6004, Centrale Nantes, France.

Abstract. A key component to the success of deep learning is the availability of massive amounts of training data. Building and annotating large datasets for solving medical image classification problems is today a bottleneck for many applications. Recently, capsule networks were proposed to deal with shortcomings of Convolutional Neural Networks (ConvNets). In this work, we compare the behavior of capsule networks against ConvNets under typical datasets constraints of medical image analysis, namely, small amounts of annotated data and class-imbalance. We evaluate our experiments on MNIST, Fashion-MNIST and medical (histological and retina images) publicly available datasets. Our results suggest that capsule networks can be trained with less amount of data for the same or better performance and are more robust to an imbalanced class distribution, which makes our approach very promising for the medical imaging community.

Keywords: capsule networks, small datasets, class imbalance.

1 Introduction

Currently, numerous state of the art solutions for medical image analysis tasks such as computer-aided detection or diagnosis rely on Convolutional Neural Networks (ConvNets) [9]. The popularity of ConvNets relies on their capability to learn meaningful and hierarchical image representations directly from examples, resulting in a feature extraction approach that is flexible, general and capable of encoding complex patterns. However, their success depends on the availability of very-large databases representative of the full-variations of the input source. This is a problem when dealing with medical images as their collection and labeling are confronted with both data privacy issues and the need for time-consuming expert annotations. Furthermore, we have poor control of the class distributions in medical databases, *i.e.* there is often an imbalance problem. Although strategies like transfer learning [14], data augmentation [12] or crowdsourcing [2] have been proposed, data collection and annotations is for many medical applications still a bottleneck [3].

ConvNets' requirement for big amounts of data is commonly justified by a large number of network parameters to train under a non-convex optimization

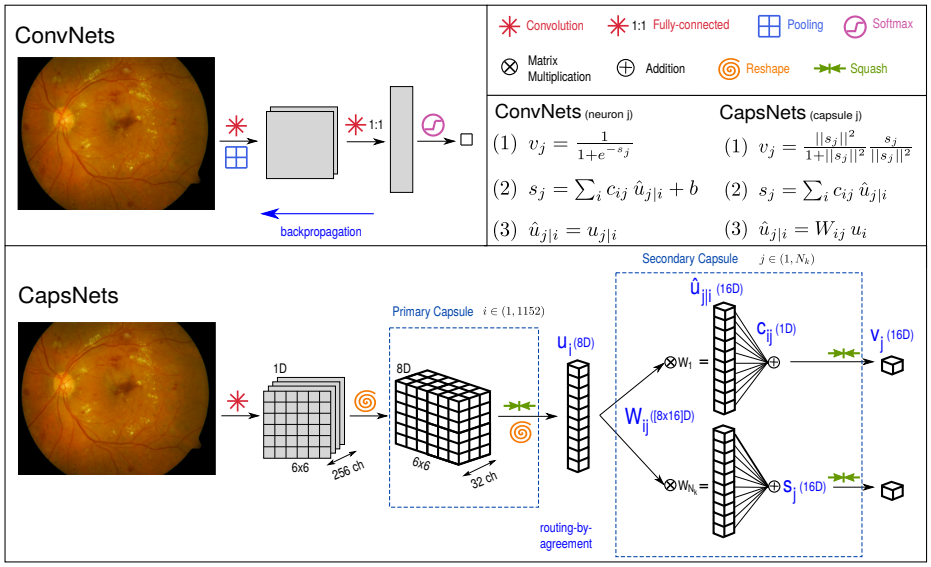

Fig. 1: Comparison of the flow and connections of ConvNets *vs.* CapsNets. Eq. (1) shows the difference between the sigmoid and squashing functions. Eq. (2) is a weighted sum of the inputs (ConvNets use bias). In CapsNets, c_{ij} are the coupling coefficients. In (3), $\hat{u}_{j|i}$ is the transformed input to the *j-th* capsule/neuron. In CapsNets, the input from the *i-th* capsule is transformed with the weights W_{ij}. While in ConvNets, the raw input from the previous neuron is used.

scheme. We argue, however, that part of these data requirements is there to cope with their poor modeling of spatial invariance. As it is known, purely convolutional networks are not natively spatially invariant. Instead, they rely on pooling layers to achieve translation invariance, and on data-augmentation to handle rotation invariance. With pooling, the convolution filters learn the distinctive features of the object of interest irrespective of their location. Thereby losing the spatial relationship among features which might be essential to determine their class (e.g. the presence of plane parts in an image does not ensure that it contains a plane).

Recently, capsule networks [10] were introduced as an alternative deep learning architecture and training approach to model the spatial/viewpoint variability of an object in the image. Inspired by computer graphics, capsule networks not only learn good weights for feature extraction and image classification but also learn how to infer pose parameters from the image. Poses are modeled as multidimensional vectors whose entries parametrize spatial variations such as rotation, thickness, skewness, *etc.* As an example, a capsule network learns to determine whether a plane is in the image, but also if the plane is located to the left or right or if it is rotated. This is known as *equivariance* and it is a property of human one-shot learning type of vision.

In this paper, we experimentally demonstrate that the equivariance properties of CapsNets reduce the strong data requirements, and are therefore very

promising for medical image analysis. Focusing on computer-aided diagnosis (classification) tasks, we address the problems of the limited amount of annotated data and imbalance of class distributions. To ensure the validity of our claims, we perform a large number of controlled experiments on two vision (MNIST and Fashion-MNIST) and two medical datasets that targets: mitosis detection (TUPAC16) and diabetic retinopathy detection (DIARETDB1). To the best of our knowledge, this is the first study to address data challenges in the medical image analysis community with Capsule Networks.

2 Methods

In the following, we focus on the image classification problem characteristic of computer-aided diagnosis systems. Our objective is to study the behavior of Capsule Networks (CapsNets) [10] in comparison to standard Convolutional Networks (ConvNets) under typical constraints of biomedical image databases, such as a limited amount of labeled data and class imbalance. We discuss the technical advantages that make CapsNets better suited to deal with the above-mentioned challenges and experimentally demonstrate their improved performance.

2.1 Capsule vs Convolutional Networks

Similar to ConvNet approaches, CapsNets build a hierarchical image representation by passing an image through multiple layers of the network. However, as opposed to the tendency towards deeper models, the original CapsNet is formed with only two layers: a first *primary caps* layer, capturing low-level cues, followed by a specialized *secondary caps*, capable of predicting both the presence and *pose* of an object in the image. The main technical differences of CapsNets w.r.t. ConvNets are:

i) Convolutions are only performed as the first operation of the *primary caps* layer, leading as usual to a series of *feature channels*.

ii) Instead of applying a non-linearity to the scalar outputs of the convolution filters, CapsNets build tensors by grouping multiple feature channels (see the grid in Fig. 1). The non-linearity, a *squashing* function, becomes also a multidimensional operation, that takes the $j-th$ vector s_j and restricts its range to the [0,1] interval to model probabilities while preserving the vector orientation. The result of the squashing function is a vector v_j, whose magnitude can be then interpreted as the probability of the presence of a capsule's entity, while the direction encodes its pose. v_j is then the output of the capsule j.

iii) The weights W_{ij} connecting the i primary capsule to the the $j-th$ secondary capsule are an affine transformation. These transformations allow learning part/whole relationships, instead of detecting independent features by filtering at different scales portions of the image.

iv) The transformation weights W_{ij} are not optimized with the regular backpropagation but with a *routing-by-agreement* algorithm. The principal idea of the algorithm is that a lower level capsule will send its input to the higher level

capsule that *agrees* better with its input, this way is possible to establish the connection between lower- and higher-level information (refer to [10] for details).

v) Finally, the output of a ConvNet is typically a softmax layer with cross-entropy loss: $\mathcal{L}_{ce} = -\sum_x g_l(x)\ log(p_l(x))$.

Instead, for every secondary capsule, CapsNet computes the margin loss for class k:

$$\mathcal{L}_k = T_k\ \max(0, m^+ - ||\mathbf{v}_k||)^2 + \lambda\ (1 - T_k)\ \max(0, ||\mathbf{v}_k|| - m^-)^2, \qquad (1)$$

where the one-hot encoded labels T_k are 1 iff an entity of class k is present and $m^+ = 0.9$ and $m^- = 0.1$, i.e. if an entity of class k is present, its probability is expected to be above 0.9 ($||\mathbf{v}_k|| > 0.9$), and if it is absent $||\mathbf{v}_k|| < 0.1$. Since the threshold is not set as 0.5, the marginal loss forces the distances of the positive instances to be close to each other, resulting in a more robust classifier. The weight $\lambda = 0.5$.

As regularization method, CapsNet uses a decoder branch composed of two fully connected layers of 512 and 1024 filters respectively. The loss of this branch is the mean square error between the input image x and its reconstruction $\hat{x}$ both of size $N \times M$,

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N \cdot M} \sum_{n=1}^{N} \sum_{m=1}^{M} (x(n,m) - \hat{x}(n,m))^2) \qquad (2)$$

The final loss, is a weighted average of the margin loss and the reconstruction loss $\mathcal{L}_{total} = \sum_{k=1}^{N_k} \mathcal{L}_k + \alpha\ \mathcal{L}_{MSE}$.

2.2 Medical Data Challenges

It is frequent for medical image datasets to be small and highly imbalanced. Particularly, for rare disorders or volumetric segmentation, healthy samples are the majority against the abnormal ones. The cost of miss-predictions in the minority class is higher than in the majority one since high-risk patients tend to be in the minority class. There are two common strategies to cope with such scenarios: i) increase the number of data samples and balance the class distribution, and ii) use weights to penalize stronger miss-predictions of the minority class.

We propose here to rely on the equivariance property of CapsNets to exploit the structural redundancy in the images and thereby reduce the number of images needed for training. For example, in Fig. 1, we can see a fundus image in which diabetic retinopathy is present. There are different patterns present in the image that could lead to a positive diagnosis. Particularly, one can find soft and hard exudates or hemorrhages. While a ConvNet would tend to detect the presence of any of these features to make a decision, CapsNet routing algorithm is instead designed to learn to find relations between features. Redundant features are collected by the routing algorithm instead of replicated in several parts of the network to cope with invariance. We claim that the above advantages directly affect the number of data samples needed to train the networks. To demonstrate

	Conv1	Pool1	Conv2	Pool2	Conv3	-	FC1	Drop	FC2	#Params.
LeNet	5×5 6 ch	2×2	5×5 16 ch	2×2	✗	-	1×1 120 ch	✗	1×1 84 ch	$60K$
Baseline	5×5 256 ch	✗	5×5 256 ch	✗	5×5 128 ch	-	1×1 328 ch	✓	1×1 192 ch	$35.4M$

	Conv1	Pool1	Conv2	Pool2	Caps1	Caps2	FC1	Drop	FC2	#Params.
CapsNet	9×9 256 ch	✗	9×9 256 ch	✗	1152 caps 8D	N_k caps 16D	1×1 512 ch	✗	1×1 1024 ch	8.2M

Table 1: Details of each of the architectures. For convolution, we specify the size of the kernel and the number of output channels. In the case of pooling, the size of the kernel. And for capsule layers, first, the number of capsules and, in the second row, the number of dimensions of each capsule.

our hypothesis we have carefully designed a systematic and large set of experiments comparing a traditional ConvNet: LeNet [7] and a standard ConvNet: Baseline from [10], against a Capsule Network [10]. We focus on comparing their performance with regard to the medical data challenges to answer the following questions:

- How do networks behave under decreasing amounts of training data?
- Is there a change in their response to class-imbalance?
- Is there any benefit from data augmentation as a complementary strategy?

To study the generalization of our claims, our designed experiments are evaluated on four publicly available datasets for two vision and two medical applications: i) Handwritten Digit Recognition (MNIST), ii) Clothes Classification (FASHION MNIST), iii) Mitosis detection, a sub-task of mitosis counting, which is the standard way of assessing tumor proliferation in breast cancer images (TUPAC16 challenge [1]), and iv) Diabetic Retinopathy, an eye disease, that due to diabetes could end up in eye blindness over time. It is detected by a retinal screening test (DIARETDB1 dataset). Next, we provide some implementation details of the compared methods.

Architectures Since research of capsules is still in its infancy, we pick the first ConvNet, LeNet [7] for a comparison. Though this network has not many parameters (approx. 60K), it is important to notice the presence of pooling layers which reduce the number of parameters and lose the spatial relationship among features. For a fairer comparison, we pick another ConvNet with similar complexity to CapsNet, in terms of training time, that has no pooling layers, which we name hereafter Baseline and was also used for comparison in [10].

LeNet has two convolutional layers of 6 and 16 filters. Kernels are of size 5x5 and stride 1. Both are followed by a ReLU and pooling of size 2x2. Next, there are two fully connected layers with 120 and 84 filters. **Baseline** is composed of three convolutional layers of 256, 256, 128 channels, with 5x5 kernel and stride of 1. Followed by two fully connected layers of size 382, 192 and dropout. In both cases, the last layer is connected to a softmax layer with cross-entropy loss. For **CapsNet [10]**, we consider two convolutional layers of 256 filters with kernel size of 9x9 and stride of 1. Followed by two capsule layers of 8 and 16 dimensions, respectively, as depicted in Fig. 1. For each of the 16-dimensional vectors that

we have per class, we compute the margin loss like [10] and attach a decoder to reconstruct the input image. Details are summarized in Table 1.

Implementation. The networks were trained on a Linux-based system, with 32 GB RAM, Intel(R) Core(TM) CPU @ 3.70 GHz and 32 GB GeForce GTX 1080 graphics card. All models were implemented using Googles Machine Learning library TensorFlow [4]. The convolutional layers are initialized with Xavier weights [4]. All the models were trained in an end to end fashion, with Adam optimization algorithm [6], using grayscale images of size 28×28. The batch size was set to 128. For MNIST and Fashion-MNIST, we use the same learning rate and weight for the reconstruction loss as [10], while for AMIDA and DIARETDB1 we reduced both by 10. If not otherwise stated, the models were trained for 50 epochs. The reported results were tested at minimum validation loss.

3 Experimental Validation

Our systematic experimental validation compares the performance of LeNet, a Baseline ConvNet and CapsNet with regard to the three mentioned data-challenges, namely the limited amount of training data, the class-imbalance, and the utility of data-augmentation. We trained in total 432 networks, using 3 different architectures, under 9 different data conditions, for 4 repetitions, and for 4 publicly available datasets. The two first datasets are the well known MNIST [8] and Fashion-MNIST [13], with 10 classes and, 60K and 10K images for training and test respectively.

For *mitosis detection*, we use the histological images of the first auxiliary dataset from the TUPAC16 challenge [1]. There are a total of 73 breast cancer images, of $2K \times 2K$ pixels each, and with the annotated location coordinates of the mitotic figures. Images are normalized using color deconvolution [11] and only the hematoxylin channel is kept. We extract patches of size 100×100 pixels that are downsampled to 28×28, leading to about 60K and 8K images for training and test respectively. The two classes are approximately class-wise balanced after sampling.

For the *diabetic retinopathy detection*, we consider DIARETDB1 dataset [5]. It consists of 89 color fundus images of size $1.1K \times 1.5K$ pixels, of which 84 contain at least mild signs of the diabetic retinopathy, and 5 are considered as normal. Ground truth is provided as masks. We enhance the contrast of the fundus images by applying contrast limited adaptive histogram equalization (CLAHE) on the lab color space and keep only the green channel. We extract patches of 200×200 pixels that are resized to 28×28. This results in about 50K and 3K images for training and test respectively. They are approximately class-wise balanced after sampling.

[4] https://www.tensorflow.org/

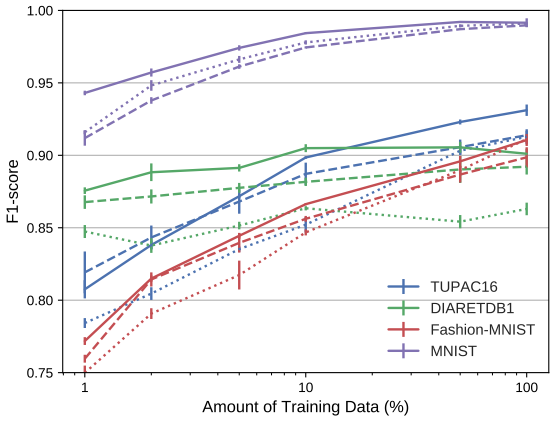

Fig. 2: Mean F_1-score and standard deviation (4 runs) for different amounts of training data. Solid line: CapsNet, dotted line: Baseline, and dashed line: LeNet.

3.1 Limited amount of training data

We compare the performance of the two networks for the different classification tasks when the original amount of training data is reduced to 50%, 10%, 5%, and 1% while keeping the original class distribution. We run each of the models for the same number of iterations that are required to train 50 full epochs using all the training data. Early-stop is applied if the validation loss does not improve in the last 20 epochs.

The results are shown in Table 2a. For almost all scenarios CapsNet performs better than LeNet and Baseline. We can observe in Figure 2 how for MNIST the gap is higher for a small amount of data and is reduced when more data is included. LeNet with 5% of the data has a similar performance to CapsNet, and better than Baseline, with 1% of the data for DIARETDB1. We attribute this behavior to the structures that are present in this type of images. All the experiments validated the significance test with a p-value < 0.05, except for those on the TUPAC16 dataset, we presume this is associated to the CapsNet limitations that we present in Section 4.

3.2 Class-imbalance

For the medical datasets, we simulate class imbalance by reducing to 20% one of the two classes. Initially, we reduce abnormal class and, afterward, the healthy class. For the other two datasets, we decrease two classes at the same time. For MNIST, we first consider reducing the classes "0" and "1" and secondly, the classes "2" and "8". Similar for Fashion-MNIST, we reduce the classes "T-shirt/top" and "Trouser", and in the second scenario, "Pullover" and "Shirt".

In Table 2b results are reported. Again, CapsNet surpasses the performance of ConvNets for all cases, except for Fashion-MNIST where the f1-scores are similar. At least one of the imbalance cases verified the significance test for all datasets.

Training Data	1%			5%			10%			50%		
	LeNet	Base.	CapsNet	LeNet	Base.	CapsNet	LeNet	Base.	CapsNet	LeNet	Base.	CapsNet
TUPAC16	**0.822**	0.784	0.809	0.872	0.835	0.872	0.890	0.852	**0.898**	0.908	0.903	**0.923**
DIARETDB1	0.870	0.847	**0.875**	0.877	0.852	**0.893**	0.883	0.863	**0.907**	0.895	0.854	**0.908**
Fashion-M.	0.759	0.749	**0.772**	0.841	0.817	**0.846**	0.856	0.847	**0.866**	0.885	0.889	**0.896**
MNIST	0.909	0.916	**0.943**	0.961	0.966	**0.975**	0.975	0.978	**0.985**	0.987	0.989	**0.992**

(a) Mean F_1-score using **different amounts of training data**.

Scenario	Balanced			Imbalanced 1			Imbalanced 2		
	LeNet	Baseline	CapsNet	LeNet	Baseline	CapsNet	LeNet	Baseline	CapsNet
TUPAC16	0.914	0.913	**0.932**	0.881	0.813	**0.892**	0.905	0.874	**0.909**
DIARETDB1	0.895	0.863	**0.899**	0.869	0.839	**0.887**	0.889	0.874	**0.898**
Fashion-M.	0.899	**0.911**	0.910	0.890	**0.902**	0.889	0.871	**0.881**	0.863
MNIST	0.989	0.991	0.991	0.988	0.989	**0.993**	0.985	0.987	**0.992**

(b) Mean F_1-score reported for different **class-imbalance** scenarios.

Data Augmentation	No			Yes		
	LeNet	Baseline	CapsNet	LeNet	Baseline	CapsNet
TUPAC16	0.904	0.892	**0.914**	0.914	0.913	**0.932**
DIARETDB1	0.883	0.864	**0.895**	0.892	0.863	**0.899**
Fashion-MNIST	0.899	**0.911**	0.910	0.902	0.911	**0.913**
MNIST	0.989	0.991	0.991	0.990	0.993	**0.994**

(c) Mean F_1-score with and wihtout **data augmentation**.

Table 2: F-1 scores under different data-challenges.

3.3 Data augmentation

In the last series of experiments, we compare the performance of the three networks using data augmentation, a common technique to increase the amount of training data and balance class distributions. The original dataset is augmented with ± 10 degrees rotations, with a translation of ± 30 pixels for medical datasets, and with flips (horizontal for Fashion-MNIST and, both horizontal and vertical for TUPAC16 and DIARETDB1). MNIST and Fashion-MNIST are augmented by 5%, for the other two datasets we consider the no augmented version to be 50% (TUPAC16) and 90% (DIARETDB1) smaller.

The performances in Table 2c show that, CapsNet *without* data augmentation achieves a similar (TUPAC16, MNIST, Fashion-MNIST) or even better (DIARETDB1) performance than ConvNets using data augmentation. All results are significant, the only Baseline for MNIST is comparable to the performance of CapsNet. These results confirm the benefits of equivariance over invariance.

4 Conclusion

In this work, we experimentally demonstrate the effectiveness of using CapsNet to improve CADx classification performance under medical data challenges. In particular, we demonstrate the increased generalization ability of CapsNets *vs.* ConvNets when dealing with the limited amount of data and class-imbalance. The performance improvement is a result of CapsNets equivariance modeling, that is, its ability to learn pose parameters along with filter weights. Together with the *routing-by-agreement* algorithm, this paradigm change requires to see

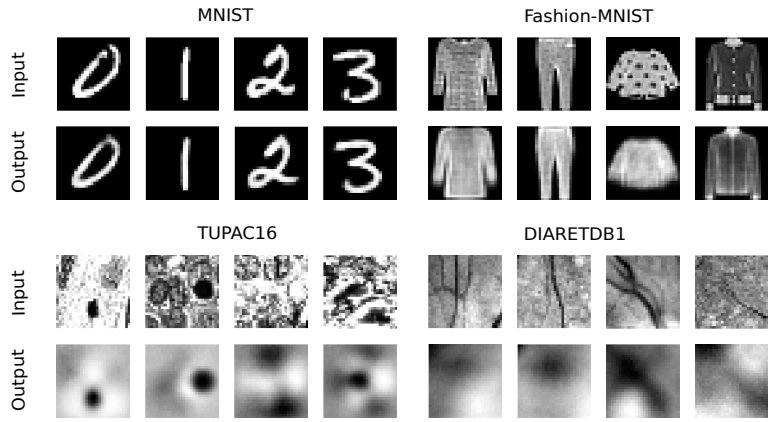

Fig. 3: Test input images and their reconstructions.

fewer viewpoints of the object of interest, and therefore fewer images, in order to learn the discriminative features to classify them. We have also reported limitations to this otherwise general improvement of CapsNets over ConvNets, their improvement in performance is significant but has a limit that we observed for the more complex TUPAC dataset at 1% (5.5K training samples).

Classification tasks where the global spatial structure plays a role can better exploit the advantages of CapsNets (DIARETDB1).

One of the disadvantages of routing-by-agreement is that is slower than regular backpropagation, CapsNet with 8.2M parameters take about the same training time per epoch than Baseline with 35.4M (a ResNet-50 has 25.6M parameters). These architectures lack purposed layers, e.g. batch normalization, that could help to ease the convergence. Depending on the number of classes, CapsNet and Baseline need between 1-3 minutes per epoch, while LeNet runs in 1-2 seconds.

Also, when visualizing the images reconstructed through the encoder-decoder branch (Fig. 3), we observe that they are blurry, especially for medical datasets with complex backgrounds. The fully-connected layers of this branch seem to be good enough to regularize the parameter optimization but lose a lot of information. Our future work includes replacing these layers with deconvolutions to get a better insight into the learned latent space.

We recommend the use of capsule networks for medical datasets where the structure is important and patterns appear in different parts of the input images, as it is for retina. Our results confirm that they perform better than standard ConvNets for the limited amount of data, at least of the order of 10k. Another potential application would be the detection of rare diseases or segmentation due to the high performance under class-imbalance.

Acknowledgment. This work has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 713673. Amelia Jiménez-Sánchez has received finan-

cial support through the "la Caixa" INPhINIT Fellowship Grant for Doctoral studies at Spanish Research Centres of Excellence, "la Caixa" Banking Foundation, Barcelona, Spain. The authors would like to thank Nvidia for the GPU donation and Aurélien Geron for his tutorial and code on Capsule Networks.

References

1. MICCAI Grand Challenge Tumor Proliferation Assessment Challenge (TUPAC16). http://tupac.tue-image.nl/, accessed: 2018-01-18
2. Albarqouni, S., Baur, C., Achilles, F., Belagiannis, V., Demirci, S., Navab, N.: Aggnet: Deep learning from crowds for mitosis detection in breast cancer histology images. IEEE Transactions on Medical Imaging 35(5), 1313–1321 (May 2016)
3. Cardoso, M.J., Arbel, T., Lee, S.L., Cheplygina, V., Balocco, S., Mateus, D., Zahnd, G., Maier-Hein, L., Demirci, S., Granger, E., Duong, L., Carbonneau, M.A., Albarqouni, S., Carneiro, G. (eds.): Intravascular Imaging and Computer Assisted Stenting, and Large-Scale Annotation of Biomedical Data and Expert Label Synthesis: 6th Joint International Workshops, CVII-STENT and Second International Workshop, LABELS (2017), held in Conjunction with MICCAI 2017
4. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Int. Conf. on Artificial Intelligence and Statistics. vol. 9, pp. 249–256. PMLR (13–15 May 2010)
5. Kalesnykiene, V., k. Kamarainen, J., Voutilainen, R., Pietil, J., Klviinen, H., Uusitalo, H.: Diaretdb1 diabetic retinopathy database and evaluation protocol
6. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR abs/1412.6980 (2014), http://arxiv.org/abs/1412.6980
7. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11), 2278–2324 (Nov 1998)
8. LeCun, Y., Cortes, C.: MNIST handwritten digit database (2010), http://yann.lecun.com/exdb/mnist/
9. Litjens, G.J.S., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A.W.M., van Ginneken, B., Sánchez, C.I.: A survey on deep learning in medical image analysis. CoRR abs/1702.05747 (2017), http://arxiv.org/abs/1702.05747
10. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 3856–3866. Curran Associates, Inc. (2017), http://papers.nips.cc/paper/6975-dynamic-routing-between-capsules.pdf
11. Vahadane, A., Peng, T., Sethi, A., Albarqouni, S., Wang, L., Baust, M., Steiger, K., Schlitter, A.M., Esposito, I., Navab, N.: Structure-preserving color normalization and sparse stain separation for histological images. IEEE Transactions on Medical Imaging 35(8), 1962–1971 (aug 2016)
12. Vasconcelos, C.N., Vasconcelos, B.N.: Increasing deep learning melanoma classification by classical and expert knowledge based image transforms. CoRR abs/1702.07025 (2017), http://arxiv.org/abs/1702.07025
13. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms (2017)
14. Zhou, J., Li, Z., Zhi, W., Liang, B., Moses, D., Dawes, L.: Using convolutional neural networks and transfer learning for bone age classification. Int. Conference on Digital Image Computing: Techniques and Applications (DICTA) pp. 1–6 (2017)