Project Report

Ferrara Luigina, Gargiulo Anna, Maruotto Ida

***Abstract*—Upon adapting the U-Net network as to perform joint intensity classification and specimen segmentation on HEp-2 cells, the network has been further modified by changing its backbone with the ResNet34 network, and adopting the Gradient Normalization algorithm for training. This paper evaluates the performances obtained on the HEP2 dataset using a combined loss - one for each task.**

I. Introduction

In recent years, deep machine learning approaches have become popular in the medical field to help discriminate between sick tissues and healthy tissues. This approaches can be used for epithelial type 2 cells, and their objective is to perform segmentation and classification of these cells. A famous network developed for image segmentation purposes is U-net, composed of an encoder part and a decoder part. The version proposed in this paper is slightly changed, allowing to perform both the image segmentation and classification at the same time, exploiting common weights coming from the encoder part of the network. This network is defined as Joint U-net [3]. The dataset considered is the HEp-2 Images Dataset, composed of Indirect ImmunoFluorescence (IIF) stained hepitelial cells for the diagnosis of Autoimmune Diseases. Indirect ImmunoFluorescence is a test for antinuclear autoantibodies (ANA) analysis. IIF slides are examined at the fluorescence microscope, and their diagnosis requires both the estimation of fluorescence intensity and the description of staining pattern. For the purpose of this project, the network should, at the same time, produce a segmented version of the images, classify the label of the images and their intensity values. For these individual tasks,

- the segmentation of the images is produced after the images have been analyzed by both the encoder and decoder part of the U-net;
- the label classification should assign each image to a specific class, corresponding to ANA staining patterns, chosen among a set of 7:
  1. homogeneous: diffuse staining of the interphase nuclei and staining of the chromatin of mitotic cells;
  2. speckled: granular nuclear staining of interphase cell nuclei;
  3. nucleolar: large coarse speckled staining within the nucleus, less than six in number per cell;
  4. centromere: several discrete speckles (40-60) distributed throughout the interphase nuclei and characteristically found in the condensed nuclear chromatin.
  5. golgi: discontinuous speckled or granular perinuclear ribbon-like staining with polar distribution in the cytoplasm [1];
  6. numem (Nuclear Membrane) homogeneous staining of the nucleus with greater intensity at its outer rim;[1]
  7. mitsp (Mitotic Spindle) the spindle fibers between the poles are stained in mitotic cells [1];

  Example of each class is shown in Fig. 1.
- the intensity classification should instead determine if the fluorescence has an intermediate (weak) or positive (strong) level.

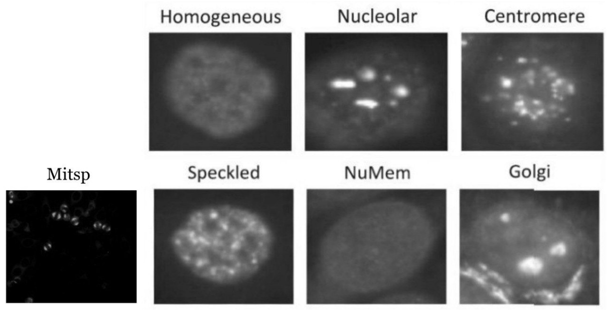

Fig. 1. Reference image for each class

II. Description of the Problem

The main focus of the paper is to describe the performances of the joint U-net on the HEP2-cells dataset when the architecture of the network is further changed.U-net takes its name from its shape, so by keeping the encoder-decoder ratio, this network can still be used for medical approaches. The encoder of the network has to be substituted with the reference module of the ResNet34, a network trained for image classification. The decoder should be symmetrical, so it is necessary to evaluate the second part of the network so that it matches the first. Since the problem is multi task (segmentation and classification of images), the approach to be used to compute the weights of the loss of the network, which is a MultiTask loss, is the Gradient Normalization Algorithm, designed to find the best weights when a network has to solve more than one problem. The estimation of the performance needs to be evaluated with two different sets of losses.

III. Methods

To fulfill the requirements of the problem, it is first necessary to define the loss for the project. The loss exploited is defined as a multi task loss, since every single task is described by a specific loss. To be thorough, the task is taken upon with two different multi task losses. For the first approach, the complete loss is defined as so:

- binary cross entropy, for the intensity classification of the images;
- cross entropy, for the label classification of the images;
- dice loss, for the segmentation of the images.

For the second approach, the loss is, instead, defined as:

- binary cross entropy, for the intensity classification of the images;
- cross entropy, for the label classification of the images;
- binary cross entropy, for the segmentation of the images.

The loss is defined as a class that instantiates the binary cross entropy, the cross entropy and the dice losses. The *forward* function returns a unique tensor, equal to the concatenation of the single tensors of the single losses.

Moreover, to train the network the Gradient Normalization algorithm [2] was necessary. The algorithm automatically balances training in deep multitask models by dynamically tuning gradient magnitudes. It also matches or surpasses the performance of exhaustive grid search methods, despite only involving a single asymmetry hyperparameter α. In this paper, we chose as a value for α 0.06.
The GradNorm can be used to train the whole network by following these steps:

Algorithm 1 Training with GradNorm

Initialize $w_i(0) = 1 \; \forall i$
Initialize network weights $\mathcal{W}$
Pick value for $\alpha > 0$ and pick the weights W (usually the final layer of weights which are shared between tasks)
for $t = 0$ **to** max_train_steps **do**
  Input batch x_i to compute $L_i(t) \; \forall i$ and $L(t) = \sum_i w_i(t) L_i(t)$ [standard forward pass]
  Compute $G_W^{(i)}(t)$ and $r_i(t) \; \forall i$
  Compute $\overline{G}_W(t)$ by averaging the $G_W^{(i)}(t)$
  Compute $L_{\text{grad}} = \sum_i |G_W^{(i)}(t) - \overline{G}_W(t) \times [r_i(t)]^\alpha|_1$
  Compute GradNorm gradients $\nabla_{w_i} L_{\text{grad}}$, keeping targets $\overline{G}_W(t) \times [r_i(t)]^\alpha$ constant
  Compute standard gradients $\nabla_{\mathcal{W}} L(t)$
  Update $w_i(t) \mapsto w_i(t+1)$ using $\nabla_{w_i} L_{\text{grad}}$
  Update $\mathcal{W}(t) \mapsto \mathcal{W}(t+1)$ using $\nabla_{\mathcal{W}} L(t)$ [standard backward pass]
  Renormalize $w_i(t+1)$ so that $\sum_i w_i(t+1) = T$
end for

Fig. 2. Training with Gradient Normalization

The approach used was to define a Gradient Normalization class where the training for each epoch is handled.

Before performing any training, the requirements for the project specified that the backbone of the network had to be changed. In Fig.3 the classic joint U net model is shown. The encoder of the network (shown in green and blue in Fig.3) is made of several *Down* blocks, each constituted of a MaxPooling layer, and a Double Convolutional Layer (a double sequence of Conv2D, Batch Normalization and ReLU layers).

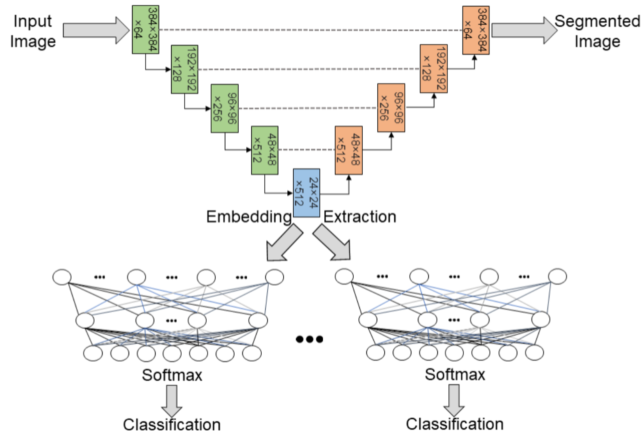

Fig. 3. Original Joint U-Net Architecture

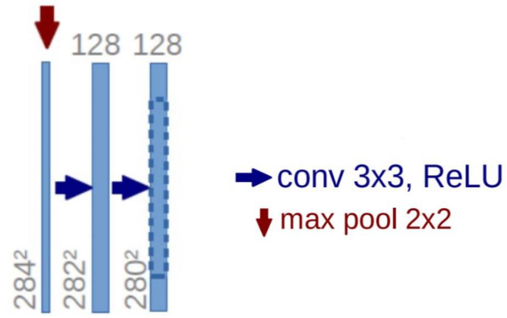

Fig. 4. Example of the Down block

To introduce the ResNet34 in the joint U-net model, it was first necessary to identify the module block of the network.

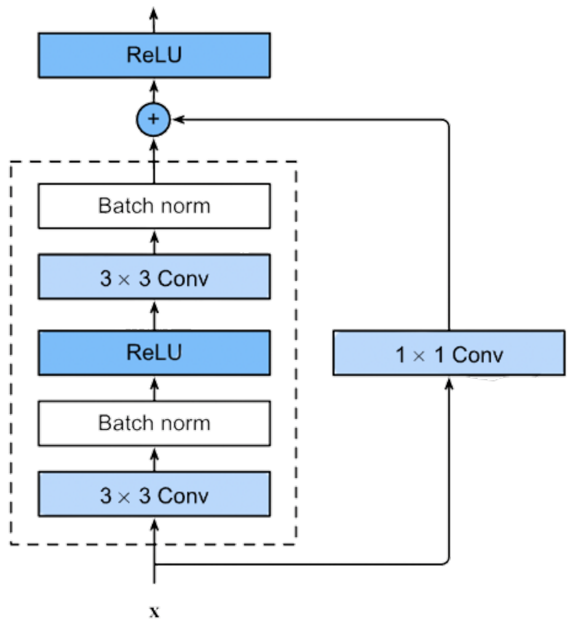

Fig. 5. ResNet34 Module Block

The block is characterized by a similar layout with respect to the original Down Block, with the addition of a skip connection that is linked to the output through a Convolutional layer of Kernel 1x1, that is needed to adapt the dimensions of the output to the original dimensions of the input.

FOLDS e training

Figures and tables should be labeled and numbered, such as in Table I and Fig. 6.

TABLE I
SIMULATION PARAMETERS

Information message length	$k = 16000$ bit
Radio segment size	$b = 160$ bit
Rate of component codes	$R_{cc} = 1/3$
Polynomial of component encoders	$[1, 33/37, 25/37]_8$

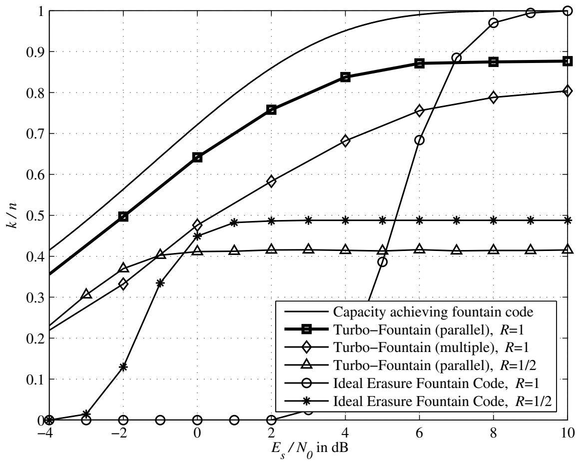

Fig. 6. Simulation results on the AWGN channel. Average throughput k/n vs E_s/N_0.

IV. FILLING THIS PAGE - RESULTS

V. CONCLUSION AND FUTURE WORK

This section summarizes the paper. References should be cited as numbers, and should be ordered by their appearance (example: "... as shown in [2], ..."). Only references that are actually cited can be listed in the references section. The references' format should be evident from the examples in this text.

References should be of academic character and should be published and accessible. You must cite all used sources. Examples of good references include text books and scientific journals or conference proceedings. If possible, citing internet pages should be avoided. In particular, Wikipedia is *not* an appropriate reference in academic reports. Avoiding references in languages other than English is recommended.

REFERENCES

[1] Information derived from *ICAP: International Consensus on ANA Patterns.* https://www.anapatterns.org/

[2] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, Andrew Rabinovich. *GradNorm*: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks https://arxiv.org/pdf/1711.02257.pdf

[3] Gennaro Percannella, Umberto Petruzzello,Pierluigi Ritrovato,Leonardo Rundo,Francesco Tortorella, Mario Vento *IEEE: Joint Intensity Classification and Specimen Segmentation on HEp-2 Images: a Deep Learning Approach* https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9956212&tag=1