

## Classification of glomerular hypercellularity using convolutional features and support vector machine

Paulo Chagas<sup>a,1</sup>, Luiz Souza<sup>a,1</sup>, Ikaro Araújo<sup>b</sup>, Nayze Aldeman<sup>c</sup>, Angelo Duarte<sup>d</sup>, Michele Angelo<sup>d</sup>, Washington L.C. dos-Santos<sup>e,\*</sup>, Luciano Oliveira<sup>a,\*</sup>

<sup>a</sup> IVISION Lab, Universidade Federal da Bahia, Bahia, Brazil

<sup>b</sup> PPGM, Universidade Federal da Bahia, Bahia, Brazil

<sup>c</sup> Departamento de Medicina Especializada – Universidade Federal do Piauí, Piauí, Brazil

<sup>d</sup> Universidade Estadual de Feira de Santana, Bahia, Brazil

<sup>e</sup> Fundação Oswaldo Cruz – Instituto Gonçalo Moniz, Bahia, Brazil

### ARTICLE INFO

#### Keywords:

Hypercellularity  
Human kidney biopsy  
Convolutional neural network

### ABSTRACT

Glomeruli are histological structures of the kidney cortex formed by interwoven blood capillaries, and are responsible for blood filtration. Glomerular lesions impair kidney filtration capability, leading to protein loss and metabolic waste retention. An example of lesion is the glomerular hypercellularity, which is characterized by an increase in the number of cell nuclei in different areas of the glomeruli. Glomerular hypercellularity is a frequent lesion present in different kidney diseases. Automatic detection of glomerular hypercellularity would accelerate the screening of scanned histological slides for the lesion, enhancing clinical diagnosis. Having this in mind, we propose a new approach for classification of hypercellularity in human kidney images. Our proposed method introduces a novel architecture of a convolutional neural network (CNN) along with a support vector machine, achieving near perfect average results on FIOCRUZ data set in a binary classification (lesion or normal). Additionally, classification of hypercellularity sub-lesions was also evaluated, considering mesangial, endocapilar and both lesions, reaching an average accuracy of 82%. Either in binary task or in the multi-classification one, our proposed method outperformed Xception, ResNet50 and InceptionV3 networks, as well as a traditional handcrafted-based method. To the best of our knowledge, this is the first study on deep learning over a data set of glomerular hypercellularity images of human kidney.

### 1. Introduction

Digital histopathology is a research field that exploits digital images for the analysis of tissue samples. The digital pictures are obtained either by scanning histological whole-slide-images (WSIs) or by collecting snapshots of histological structures relevant for the diagnosis of diseases [2]. This approach makes gathering large-scale data sets of histological lesions easier to review or to exchange information among pathologists without the inconvenience of working with the actual glass slides. The evolution of the computer vision field impacted the entire digital medicine, supporting pathologists on the automatic analysis of various types of medical images, as well as improving the accuracy of computer-aided diagnosis [28,27].

In the special case of renal histopathology, disease markers are mostly found in the glomeruli, presenting highly diverse and heterogeneous characteristics. The glomerulus is a histological structure from the kidney cortex, formed by a network of capillaries charged of performing blood filtration. As an elementary filtering structure, it is targeted with many primary and systemic diseases, leading to different patterns of glomerular lesions. Finding and classifying glomerular lesions are fundamental steps toward the diagnosis of many kidney diseases. These tasks rely on the expertise of pathologists and much effort has been made to better define and create consensus about relevant lesions. In fact, after successive discussion and validation studies in the field, increased consistency has been achieved in the diagnosis and classification of glomerular renal diseases such as lupus nephritis, IgA

\* Corresponding authors.

E-mail addresses: [paulo.chagas@ufba.br](mailto:paulo.chagas@ufba.br) (P. Chagas), [luiz.otavio@ufba.br](mailto:luiz.otavio@ufba.br) (L. Souza), [ikaroa@ufba.br](mailto:ikaroa@ufba.br) (I. Araújo), [nayze.aldeman@iesvap.edu.br](mailto:nayze.aldeman@iesvap.edu.br) (N. Aldeman), [angeloduarte@uefs.br](mailto:angeloduarte@uefs.br) (A. Duarte), [mfangelo@ecomp.uefs.br](mailto:mfangelo@ecomp.uefs.br) (M. Angelo), [wluis@bahia.fiocruz.br](mailto:wluis@bahia.fiocruz.br) (W.L.C. dos-Santos), [lrebouca@ufba.br](mailto:lrebouca@ufba.br) (L. Oliveira).

<sup>1</sup> These authors contributed equally to this work.

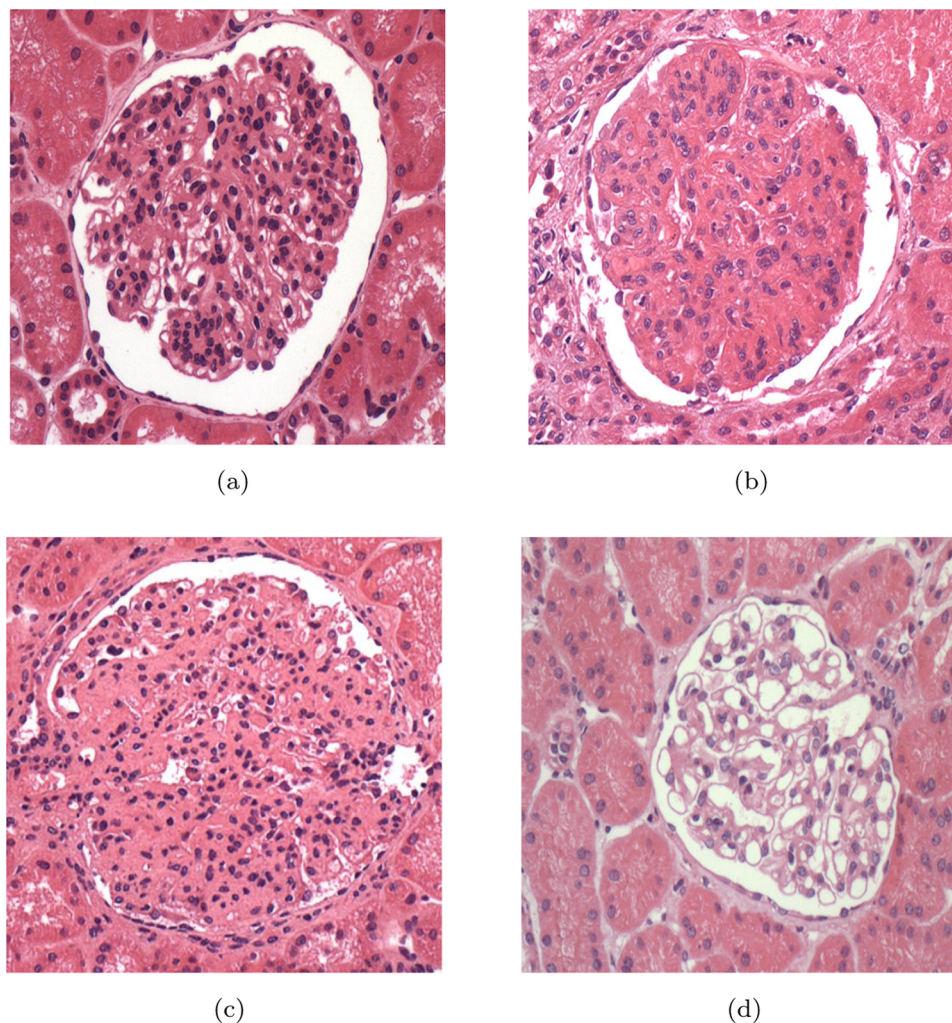


Fig. 1. Four glomeruli containing (a) endocapillary, (b) mesangial, (c) both lesion, and (d) no lesion.

nephropathy, and rejection of kidney transplant [3,39,23]. Some limiting factors to the performance of histological diagnosis are the complexity of lesions, which, in some cases, may impair a clear definition in terms of criteria and consequently a suitable agreement among pathologists [4].

Particularly, glomerular hypercellularity is a frequent lesion found in kidney biopsies, defined by an increase in the number of cells in the glomeruli. This type of lesion is an integral component of many glomerular diseases such as proliferative and membranoproliferative glomerulonephritis, being a marker of activity in lupus and IgA nephropathy [3,39]. Hypercellularity can be identified by a careful look at the histological sections from the glomeruli, searching for the presence of agglomerates formed by four or more cell nuclei in the mesangial area (mesangial hypercellularity), or by cell aggregates that fill the capillary lumen (endocapillary hypercellularity) [10,14]. Fig. 1 shows the complexity of this problem, illustrating the two types of hypercellularity, including the cases with both and none lesions.

Although hypercellularity is easy to define and usually easy to be assessed in histological sections, an agreement among pathologists may decrease for focal hypercellularity and for occurrences in specific regions of the glomerulus. For instance, a recent report from the IgA Nephropathy Classification Working Group showed inconsistencies among specialist even in the use of dichotomous MEST system scores such as E (endocapillary hypercellularity) and M (mesangial hypercellularity) [39]. Correct assessment of these scores is crucial for relevant clinical-pathological correlation, as well as for predicting the patient outcome.

A consistent glomerulus classification can be deemed as an important and difficult step towards diagnosing a renal disease in a biopsy evaluation [30].

Some works have already approached the tasks of glomerulus identification and segmentation [32,24,35], which are useful in situations that require an analysis of the entire WSI. Barros et al. [5] proposed a method relying on classical image pre-processing techniques and a k-nearest neighborhood to classify hypercellularity lesions; that work used 811 images of human glomeruli (referred here as FIOCRUZ data set) stained with hematoxylin-eosin (H&E) and periodic acid-Schiff (PAS) from a set of biopsy slides. More recently, deep neural networks outperformed handcrafted features for some tasks on histological images as well, achieving stunning results in different scenarios [22,41,34,40,37,20,42,13,16]. In particular to glomerular detection with deep-learning, Marsh et al. [29] introduced a convolutional neural network for automatic localization of glomeruli, further classifying global glomerulosclerosis in donor kidney biopsies for transplantation.

An automated process for glomerular lesion classification would have many applications, such as: large-scale classification of cases based on histological images, consistency of morphological classification, and identification of tissue markers of disease progression.

### 1.1. Contributions

Three main contributions are brought here: (i) Instead of using conventional classification methods as in [5], we propose a CNN-based

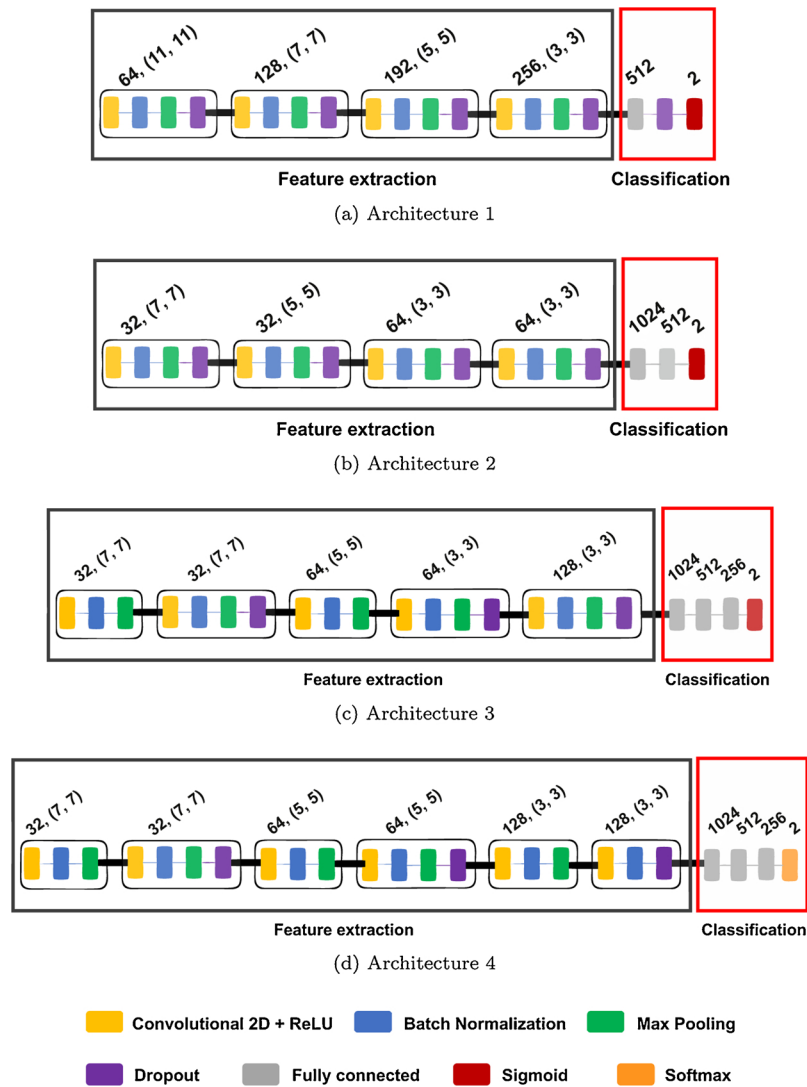


Fig. 2. Four CNN architectures proposed here: (a) architecture 1 and (b) architecture 2, with four convolutional layers in the backbone; (c) architecture 3 and (d) architecture 4, with five and six convolutional layers, respectively, in the backbone.

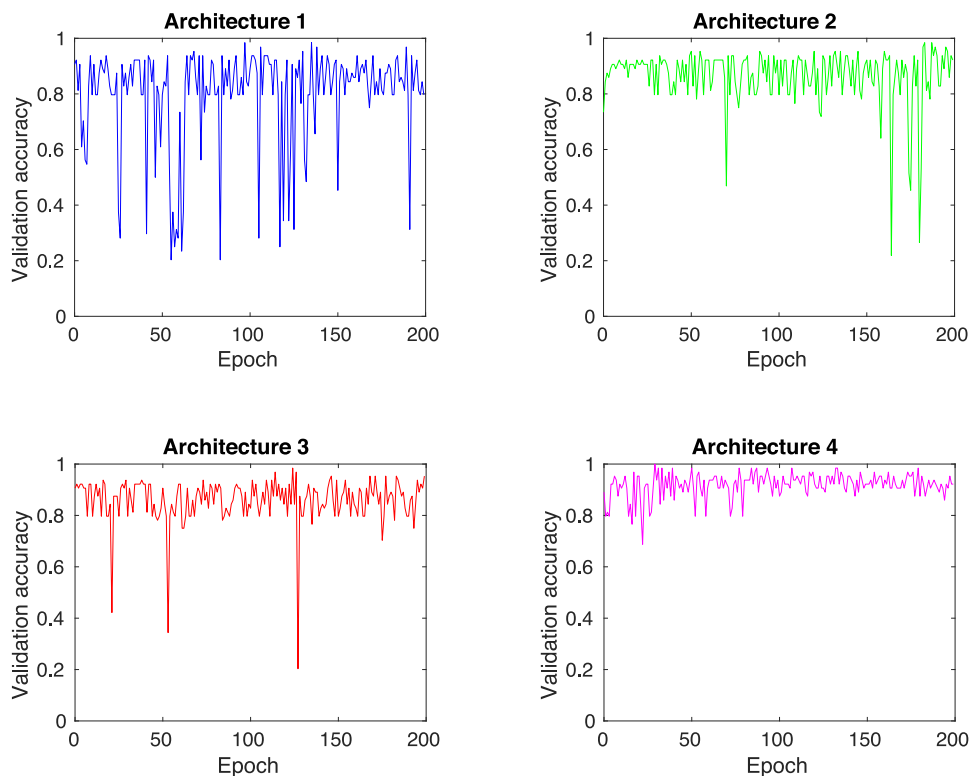
architecture to extract trainable features to represent a glomerulus, (ii) by using the proposed CNN as a feature extractor, an SVM classifies the CNN features as a normal or an injured glomerulus, (iii) we also extend the proposed model for classification of specific hypercellularity lesions (endocapillary hypercellularity, mesangial hypercellularity, and both), providing an analysis of the generated features for both binary and multi-lesion classification. The final CNN-SVM classifier reached near perfect results in four different train/test splits of the data set introduced in [5]; in the multi-classification task, the same architecture achieved an average performance of 82% in 10-fold cross-validation, surpassing Xception, ResNet50 and InceptionV3 networks, as well as the method propose in [5], this latter modified to deal with multi-class.

## 2. Classifying glomerular hypercellularity

The classification of a glomerular hypercellularity lesion could be tackled as defining areas and counting nuclei. If the number of nuclei per area surpasses a threshold, one can diagnose a glomerulus as with a hypercellularity lesion. Instead of following this pathologist-annotation approach, an automatic classification consists of using examples of histological images to train a classifier. A histological image is a 2-dimensional grid of pixels that brings specific information such as colors, edges, shapes, textures, which can be general or specific to classify a

glomerular lesion. Consequently, conceiving a successful feature extractor demands some domain expertise, which brings us to the following question: *What is the best feature set for classifying glomerular hypercellularity lesions?*

Many feature extraction techniques are available in the literature, and a specific method could be designed as well. In contrast to conventional classifiers, deep-learning aims to automatically learn hierarchical feature representations of the input data, without the need of creating any particular feature extractor [26]. Our work proposes a novel CNN-based architecture for glomerular hypercellularity classification. After training a CNN, it is possible to use a strong classifier on the convolutional backbone of features. This way, we propose to use a CNN architecture to extract trainable features, which ultimately will feed an SVM to carry on the final classification. The proposed architecture is evaluated for both binary and multi-class classification. The rationale to use an SVM is based on the main characteristic of this classifier that is to cast optimization problems, which are convex and quadratic. Ultimately, these characteristics guarantee that the hyper-plane found is the optimum one. The second reason is to analyze the behavior of feature space extracted from the CNN, which empirically demonstrated to be linear, in our experiments. Linearity in the feature space is expected to provide faster and higher results.



**Fig. 3.** Stability evaluation of CNN architectures. From ‘architecture 1’ to ‘architecture 4’, accuracy reaches stability as the number of training epochs increases. Best results were achieved with ‘architecture 4’, which used a softmax layer at the top, resulting in a more stable accuracy on the validation set.

### 2.1. Conceiving the proposed CNN architecture

There are several well-established CNN architectures available in the literature [6], which were designed to be robust to deal with hundreds of different classes. However, these models tend to overfitting, when trained using few data. Since the data set we used [5] consists of a small training set, we decided to build our own architecture from scratch, modifying it accordingly to our needs. The ultimate goal is to focus on achieving a high accuracy, avoiding overfitting.

A CNN architecture is organized in layers, each one applying a specific operation. Although there are many variations of CNN architectures, they share some basic components, such as convolutional, pooling, and fully-connected layers [17]. The convolutional layer is the fundamental building block of a CNN model, which is comprised of various learnable kernels (filters) followed by a nonlinear activation function. A pooling layer (usually applied after a convolutional layer) is used to compute feature maps condensed in a smaller representation with the goal of achieving some invariance. After some convolutional and pooling operations, the top of the network results in a high-level representation of the input image, which is more robust than the raw pixel information, or hopefully than handcrafted features. This type of architecture requires a fully-connected layer to perform high-level classification using those features, working as a multilayer perceptron (MLP) on top of a CNN backbone.

Four architectures were initially implemented and Fig. 2 highlights the convolutional blocks (CNN backbone) used for feature extraction, and the MLP blocks (fully-connected layers and final activation) used for classification. The first architecture was designed in the view of investigating how the lesion classification behaved using fewer layers. In addition to the operations previously cited, batch normalization, regularization, and dropout operations were applied to reduce overfitting. The first architecture (Fig. 2a) is composed mainly of four convolutional layers, with the other operations applied between those layers, followed by one fully-connected layer. A rectified linear unit

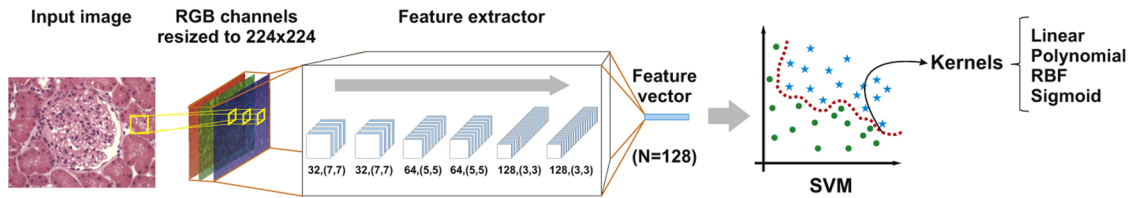
(ReLU) was used as an activation function and max-function for pooling operations. For the calculation of the class probabilities after the fully-connected layers, a sigmoid function was first tried, and further changed to a soft-max function. With this first architecture in mind, updates were performed based on the stability of the accuracy curve in the validation set, and other three architectures were proposed (Fig. 2b–d).

In order to choose the best model among the candidate architectures, we randomly selected 90% of the data set for training the model, while using 10% for validation. To deal with the great size of the data set in memory, we applied a mini-batch strategy, which consists of using several batches of  $N$  images to update the final model (instead of one single block of data). After each epoch, the proposed architecture was evaluated by using the validation set. Since we focused on reducing the overfitting, the more likely architecture to be selected would be the one with high accuracy and less oscillation in the accuracy. Fig. 3 shows the accuracy curve for each architecture, illustrating the raise not only on the accuracy peak, but also on the stability of the curve after several epochs. Our final CNN architecture (Fig. 2d) consists mainly of six convolutional layers, five max-pooling layers, followed by three fully-connected and one soft-max layers for classification. The training parameters were empirically obtained through several experiments on the four architectures. The best results using **architecture 4** were achieved by training the deep network using the following parameters: 200 epochs, Adam training algorithm [25],  $10^{-6}$  of decay rate, batch size of 32, and a learning rate of  $10^{-4}$ .

### 2.2. Classifying the CNN features with SVM

After choosing the best architecture, the trained CNN features fed an SVM, instead of the multi-layer perceptron used for training the model. This CNN-SVM architecture was evaluated with four kernel functions: Linear, radial basis function (RBF), polynomial and sigmoid (see Fig. 4).

SVM is a supervised binary classifier, which finds an optimal



**Fig. 4.** CNN-SVM architecture. From left to right: a glomerulus as an **input image** in an RGB color space, **resized to 224 × 224** pixels. After applying architecture 4 (Fig. 2d) for **feature extraction**, a **feature vector** with 128 features is generated. Finally, the resultant feature vector is classified by an **SVM** evaluated by considering **linear, polynomial, RBF and sigmoid** kernel functions.

hyperplane to separate the classes of hypercellularity from those of normal glomeruli by using  $\nu$  support vectors. When these classes are non-linearly separable, different kernel functions can be used to map the input vectors to a higher-dimensional space (so-called *feature space*), in which the input image can be linearly separated. To classify an input feature vector, SVM evaluates the sign of a function  $f(x)$ , given by

$$f(x) = \text{sign} \left( \sum_{i=1}^{\nu} y_i \alpha_i N(x_i, x_j) + b \right), \quad (1)$$

where there are  $\nu$ -support vectors with the model parameters,  $y_i$ , and a bias parameter,  $b$ , Laplacian coefficients from the dual optimization problem,  $\alpha_i$ .  $N(x_i, x_j)$  is a kernel function.

### 3. Experimental analysis

#### 3.1. Data set

In order to assess the performance of our proposed CNN architecture, the data set introduced in [5] was used. The data set consists of 811 images, containing 300 images of normal human glomerulus, while 511 images of human glomerulus with hypercellularity. As the images originated from human kidneys with different diseases, the cellular component of the hypercellularity varies among the cases. The images were selected from the digital histological image library of the Gonalo Moniz Institute (FIOCRUZ), including cases of all the kidney biopsies performed for the diagnosis of glomerular diseases in referral nephrology services of public hospitals in Bahia state, Brazil, between 2003 and 2015. The tissue samples were fixed in Bouin's fixative or formalin-acetic acid-alcohol, included in paraffin. Sections of 2–3  $\mu\text{m}$  were stained by H&E and PAS. The images were captured using an Olympus QColor 3 digital camera attached to a Nikon E600 optical microscope (using 200 $\times$  magnification). Details of the clinical and demographic characteristics of the patients from which kidney biopsies were collected are presented in [5]. Due to the requirements of the research protocol approved by the Ethical committee for research with human subjects, the images used for building the data set used in this study were disconnected to the patient information. Therefore, it is not possible to say anything about the number of images collected from the biopsy of a given patient.

Considering Oxford MEST, the former binary data set was relabeled into four classes: Endocapillary (90 images with endocapillary hypercellularity), mesangial (238 images with mesangial hypercellularity), endoMes (179 images with both lesions) and normal (304 images with no lesion). In this re-evaluation process using the MEST criteria for hypercellularity, four images were misclassified as lesioned glomeruli in the original binary data set used by Barros et al. [5]. This occurrence led to a difference between the number of normal glomeruli on the binary corpus (300 images) and on the 4-class (304) data set.

#### 3.2. Methodology

Most of the images have one centered glomeruli only with different dimensions, varying in height from 205 to 1333 pixels, and in width from 238 to 1459 pixels. In order to normalize the images, all images

were resized to 224 × 224 pixels (which is a common input dimension on several deep learning architectures from the literature) and divided by 255. For a comparative evaluation considering a binary classification, a  $K$ -fold cross-validation was applied, varying  $K$  as 2, 3, 5 and 10 folds. On each iteration, 1 different fold is used for validation, and the rest ( $K - 1$  folds) is used for training the model. With the best CNN architecture, we compared the performance of two types of classifiers on the top of CNN backbone: CNN-MLP and CNN-SVM. Our methodology can be summarized in two steps:

- **CNN-MLP:** the best architecture is first found by using only 90/10 split without cross-validation. Next, using different values of  $K$ , we applied  $K$ -fold cross-validation, analyzing the performance of the models using different sizes of training and validation data.
- **CNN-SVM:** For each value of  $K$  (folds), we selected the best CNN-MLP model. Then, we used the CNN features, obtained from the last layer before the fully-connected MLP, for the input of the SVM (see Fig. 4).

Finally, for multi classification, we used the same approach as the binary classification, but without varying the value of  $K$ . Since the 4-class data set is derived from the original data set used for binary classification, the number of images per class became smaller. This way, we decided to use  $K = 10$  in order to avoid a very small number of training samples per class. The one-versus-all technique was used to achieve SVM multi-class outputs. In addition to the comparison with Barros et al. [5], we also compared the proposed methods (CNN-MLP and CNN-SVM) with other three state-of-the-art CNN models: Xception [8], ResNet50 [18] and InceptionV3 [38].

**Implementation details:** The CNN models were implemented using Tensorflow [1] and Keras<sup>2</sup> on a Python environment. We used the Scikit-learn [31] implementation of the SVM for the CNN-SVM architecture. To achieve a more robust training for comparing the CNN models, we use the training procedures proposed by He et al. [19]. We used label smoothing, warm-up and cosine decay for the learning rate schedule, 200 epochs, batch size of 32, learning rate base equal to  $0.1 * \text{batch\_size} / 256 = 0.125$ , and Adam optimizer, running on a machine with 8GB RAM and an NVIDIA GEFORCE GTX 1060. Since the warm-up approach is more suitable to fine-tuning pre-trained models, we used this technique only on the Xception, ResNet50 and InceptionV3 models, which had pre-trained weights loaded from Imagenet data set [11]. These pre-trained weights were fine-tuned on the top layers only, aiming to keep the rich features learned previously.

#### 3.3. Evaluation metrics

Four metrics were used to evaluate our proposed method: **Precision (P)** as the ratio of correctly predicting glomerular hypercellularity, and the sum of predicted true positive and false positive observations (whereby high precision is regarded to low false positive rate), **recall (R)** as the ratio of correctly predicting glomerular hypercellularity, and the sum of predicted true positive and false negative observations

<sup>2</sup> <https://www.keras.io>.

**Table 1**

Comparison between four different train/test splits with CNN-MLP on binary classification. The average metrics and their standard deviations are given for precision ( $\mu P$ ), recall ( $\mu R$ ), f1-score ( $\mu F1$ ) and accuracy ( $\mu Acc$ ).

Split	CNN-MLP			
	$\mu P$	$\mu R$	$\mu F1$	$\mu Acc$
90/10	<b>0.996</b> ( $\pm 0.009$ )	<b>0.997</b> ( $\pm 0.006$ )	0.995( $\pm 0.012$ )	<b>0.996</b> ( $\pm 0.008$ )
80/20	0.995( $\pm 0.008$ )	0.994( $\pm 0.009$ )	<b>0.996</b> ( $\pm 0.006$ )	0.995( $\pm 0.007$ )
67/33	0.995( $\pm 0.005$ )	0.994( $\pm 0.005$ )	0.995( $\pm 0.005$ )	0.995( $\pm 0.005$ )
50/50	0.987( $\pm 0.003$ )	0.987( $\pm 0.003$ )	0.987( $\pm 0.003$ )	0.988( $\pm 0.003$ )

The bold values signifies the higher results found in the comparison, for all splits.

**Table 2**

Range of parameters to be evaluated for each SVM kernel.

Kernel	Function $N(x_i, x_j)$	Parameter
Linear	$x_i^T x_j$	'C': [0.001, 0.01, 0.1, 1, 10, 100]
RBF	$\exp(-\gamma   x_i - x_j  ^2)$ , where $\gamma$ refers to gamma	'C': [0.001, 0.01, 0.1, 1, 10, 100], 'gamma': [0.001, 0.01, 1, 1.5, 2]
Polynomial	$(\gamma(x_i^T x_j) + r)^d$ , where $\gamma$ denotes gamma, $r$ by coef $\theta$ and $d$ by degree	'C': [0.001, 0.01, 0.1, 1, 10, 100], 'gamma': [0.001, 0.01, 1, 1.5, 2], 'degree': [1, 2, 3, 4]
Sigmoid	$\tanh(\gamma(x_i^T x_j) + r)$ , where $\gamma$ denotes gamma and $r$ is specified by coef $\theta$	'C': [0.001, 0.01, 0.1, 1, 10, 100], 'gamma': [0.001, 0.01, 1, 1.5, 2]

The bold values signifies the higher results found in the comparison, for all splits.

(whereby high recall is regarded to low false negative rate), **f1-score (F1)** as the weighted average of precision and recall (whereby high f1-score is regarded to high precision and recall rates), and, finally, **accuracy (Acc)** as the ratio of correctly predicting glomerular hypercellularity and normal glomeruli, and the total sum of positive and negative observations (whereby accuracy is proportional to true positive and true negative rates, and inversely proportional to false positive and false negative rates). Since we applied  $K$ -fold cross-validation, we analyzed the models by using the weighted average of the metrics and their standard deviations, notated with  $\mu$  before each metric notation:  $\mu P$ ,  $\mu R$ ,  $\mu F1$ , and  $\mu Acc$ .

### 3.4. Evaluating the proposed CNN model for binary classification

The final CNN was evaluated by using the average of the chosen metrics, observing how the model generalized the classes as the size of the training and validation set changed. It is noteworthy that a  $K$  equals to 2 represents a split of 50/50, as well as,  $K$  equals to 3, 5 and 10 represents 67/33, 80/20 and 90/10, respectively. Since the training set decreases proportionally to  $K$ , we used a technique of online image augmentation, applying pre-defined random modifications (such as rotation, horizontal flip, zoom and shift) on each batch. For each value of  $K$ , there were  $K$  different validation sets, resulting in  $K$  training processes and  $K$  candidate models at the ending of the training. For example, for  $K = 10$ , there is one model for each training set combination, resulting in 10 models. When we evaluate only the CNN-MLP approach, the average of the metrics were computed with respect to these 10 models. However, since the aim was using the model as a feature extractor backbone, the best one out of the 10 candidates was selected, choosing the one with highest accuracy of all epochs. **Table 1** shows the results of training the proposed CNN-MLP model, displaying the average metrics and their standard deviations for each train/test split. In general, all the train/test splits returned top results, achieving accuracies between 98.8% (50/50 split) and 99.6% (90/10 split). As expected, in the experiments using larger training sets (90/10 split), better results were achieved, although the worst scenario (50/50 split) still showed superior values for all the proposed metrics (around 98%) in comparison with previous work [5] (85%). Another observation is the small standard deviation on all results, demonstrating the stability of the model.

### 3.5. Choosing the best SVM kernel for binary classification

Choosing optimal parameter values for the SVM kernel raises some questions about the interpretation of the model generated by this function and the results obtained. These questions were investigated in several works [7,12,21,15,36]. As shown in **Table 2**, the CNN-SVM architecture was evaluated with three parameters of kernel functions: 'C', 'gamma' and 'degree'. The regularization parameter 'C' is 1 by default, common to all SVM kernels, trading off misclassification of training examples against flatness of the solution. A low 'C' makes the classifier flatness smooth, while a high one can lead to overfitting. The 'gamma' parameter is usually 1 by default divided by number of features, and it is presented in all SVM kernels, but the linear. A small 'gamma' value represents a Gaussian distribution of the kernel function with large variance in such a way that the model might not capture the

**Table 3**

The best results per SVM kernel on binary classification.

Split	Kernel	Parameters	$\mu Acc$
90/10	<b>Linear</b>	'C': <b>1</b>	<b>1.000</b> $\pm$ (0.000)
	<b>RBF</b>	'C': <b>0.1</b> , 'gamma': <b>0.001</b>	<b>1.000</b> $\pm$ (0.000)
	<b>Polynomial</b>	'C': <b>1</b> , 'degree': <b>1</b> , 'gamma': <b>1</b>	<b>1.000</b> $\pm$ (0.000)
	<b>Sigmoid</b>	'C': <b>0.01</b> , 'gamma': <b>0.01</b>	<b>1.000</b> $\pm$ (0.000)
80/20	Linear	'C': 0.001	0.994 $\pm$ (0.011)
	<b>RBF</b>	'C': <b>0.1</b> , 'gamma': <b>0.01</b>	<b>0.996</b> $\pm$ (0.010)
	Polynomial	'C': 0.001, 'degree': 1, 'gamma': 1	0.994 $\pm$ (0.011)
	<b>Sigmoid</b>	'C': <b>0.01</b> , 'gamma': <b>0.01</b>	<b>0.996</b> $\pm$ (0.010)
67/33	Linear	'C': 10	0.993 $\pm$ (0.006)
	<b>RBF</b>	'C': <b>1</b> , 'gamma': <b>1</b>	<b>0.994</b> $\pm$ (0.003)
	Polynomial	'C': 0.001, 'degree': 3, 'gamma': 2	0.994 $\pm$ (0.007)
	Sigmoid	'C': 1, 'gamma': 0.01	0.991 $\pm$ (0.009)
50/50	<b>Linear</b>	'C': <b>0.01</b>	<b>0.988</b> $\pm$ (0.005)
	<b>RBF</b>	'C': <b>10</b> , 'gamma': <b>0.01</b>	<b>0.988</b> $\pm$ (0.005)
	<b>Polynomial</b>	'C': <b>0.001</b> , 'degree': <b>2</b> , 'gamma': <b>1.5</b>	<b>0.988</b> $\pm$ (0.005)
	<b>Sigmoid</b>	'C': <b>1</b> , 'gamma': <b>0.01</b>	<b>0.988</b> $\pm$ (0.005)

The bold values signifies the higher results found in the comparison, for all splits.

**Table 4**

Comparison between four different train/test splits with CNN-SVM on binary classification. The average metrics and their standard deviation are given for average precision ( $\mu P$ ), average recall ( $\mu R$ ), average f1-score ( $\mu F1$ ) and average accuracy ( $\mu Acc$ ).

Split	CNN-SVM			
	$\mu P$	$\mu R$	$\mu F1$	$\mu Acc$
90/10	<b>1.000(±0.000)</b>	<b>1.000(±0.000)</b>	<b>1.000(±0.000)</b>	<b>1.000(±0.000)</b>
80/20	0.996(±0.006)	0.996(±0.007)	0.996(±0.003)	0.996(±0.010)
67/33	0.996(±0.004)	0.996(±0.004)	0.996(±0.001)	0.994(±0.007)
50/50	0.988(±0.008)	0.983(±0.015)	0.985(±0.004)	0.988(±0.005)

The bold values signifies the higher results found in the comparison, for all splits.

**Table 5**

Overall comparison between CNN-MLP, CNN-SVM, Xception, ResNet50, and IceptionV3 on binary classification.

Split	Model	$\mu P$	$\mu R$	$\mu F1$	$\mu Acc$
90/10	Xception	0.995(±0.007)	0.995(±0.008)	0.995(±0.008)	0.995(±0.008)
	ResNet50	0.991(±0.010)	0.991(±0.011)	0.991(±0.011)	0.991(±0.011)
	InceptionV3	0.990(±0.010)	0.990(±0.010)	0.990(±0.010)	0.990(±0.010)
	CNN-MLP	0.996(±0.009)	0.997(±0.006)	0.995(±0.012)	0.996(±0.008)
	CNN-SVM	<b>1.000(±0.000)</b>	<b>1.000(±0.000)</b>	<b>1.000(±0.000)</b>	<b>1.000(±0.000)</b>
	[5] <sup>a</sup>	0.923	0.880	0.901	0.883(±0.036)
80/20	Xception	0.991(±0.006)	0.991(±0.006)	0.991(±0.006)	0.991(±0.006)
	ResNet50	0.989(±0.007)	0.988(±0.007)	0.988(±0.007)	0.988(±0.007)
	InceptionV3	0.986(±0.007)	0.986(±0.007)	0.986(±0.007)	0.986(±0.007)
	CNN-MLP	0.995(±0.008)	0.994(±0.009)	0.996(±0.006)	0.995(±0.007)
	CNN-SVM	<b>0.996(±0.006)</b>	<b>0.996(±0.007)</b>	<b>0.996(±0.003)</b>	<b>0.996(±0.010)</b>
	[5]	0.918(±0.051)	0.845(±0.070)	0.877(±0.035)	0.852(±0.039)
67/33	Xception	0.986(±0.004)	0.986(±0.004)	0.986(±0.004)	0.986(±0.004)
	ResNet50	0.985(±0.003)	0.985(±0.003)	0.985(±0.003)	0.985(±0.003)
	InceptionV3	0.984(±0.001)	0.984(±0.001)	0.984(±0.001)	0.984(±0.001)
	CNN-MLP	0.995(±0.005)	0.994(±0.005)	0.995(±0.005)	0.995(±0.005)
	CNN-SVM	<b>0.996(±0.004)</b>	<b>0.996(±0.004)</b>	<b>0.996(±0.001)</b>	<b>0.994(±0.007)</b>
	[5]	0.911(±0.023)	0.817(±0.082)	0.859(±0.046)	0.834(±0.045)
50/50	Xception	0.972(±0.003)	0.971(±0.003)	0.971(±0.003)	0.971(±0.003)
	ResNet50	0.980(±0.002)	0.980(±0.002)	0.980(±0.002)	0.980(±0.002)
	InceptionV3	0.980(±0.004)	0.980(±0.004)	0.980(±0.005)	0.980(±0.004)
	CNN-MLP	<b>0.987(±0.003)</b>	<b>0.987(±0.003)</b>	<b>0.987(±0.003)</b>	<b>0.988(±0.003)</b>
	CNN-SVM	0.988(±0.008)	0.983(±0.015)	0.985(±0.004)	0.988(±0.005)
	[5]	0.908(±0.075)	0.741(±0.189)	0.793(±0.085)	0.775(±0.063)

<sup>a</sup>  $\mu P$ ,  $\mu R$  and  $\mu F1$  are given, but not their standard deviations in the article. We just list the values in this row as they are in [5].

“shape” of the data set. When ‘gamma’ is high, the resulting model will behave similarly to a linear kernel with a set of hyperplanes separating the points of the two classes; hence, large gamma takes to high bias and low variance models, and vice-versa. The ‘degree’ parameter is 3 by default, and used only in polynomial kernel function. This parameter adjusts the feature space for higher-dimensional interactions. Larger ‘degrees’ tend to overfit the data.

The same range of  $K$  values applied to evaluate the CNN was also used to evaluate SVM. Table 3 shows the best parameter combinations for each kernel at each split, using accuracy as a metric for optimization. It is noteworthy that the linear kernel achieving top results means that the feature space can be linearly separable. The overall results of the CNN-SVM approach are summarized in Table 4, showing the performance of the proposed approach using the previously defined metrics. For the 90/10 split, all SVM kernels showed perfect  $\mu Acc$ . For the 80/20 split, RBF and sigmoid kernels achieved the highest results. In the 67/33 split, RBF kernel obtained the best result. In the 50/50 split, all SVM kernels achieved the same results. Table 5 brings a detailed comparison between the two proposed approaches and the three state-of-the-art CNN models along with the method proposed by Barros et al. [5]. As that previous work [5] did not use the F1-score for evaluation, we calculated this score based on the precision and recall provided on the cross-validation stage. Hence, we could compare the three results using all four metrics, considering 10-fold cross-validation (90/10

split). We used their method to calculate the scores for the other splits. We can note that the CNN-MLP already outperforms the three deep models in all train/test splits for all metrics. CNN-SVM achieved the highest results for all train/test splits except for 50/50, but still outperformed the three deep models in all splits.

### 3.6. Extending the proposed architecture to multi-classification

We also proposed the use of our CNN-SVM architecture for classification of a 4-class data set, including the following classes: Endocapillary hypercellularity, mesangial hypercellularity, *endoMes* (both lesions) hypercellularity, and normal glomerulus. The same

**Table 6**

The best parameters per SVM kernel on 4-class classification.

Kernel	Parameters	$\mu Acc$
Linear	‘C’: 0.01	0.815
RBF	‘C’: 1, ‘gamma’: 0.01	0.816
<b>Polynomial</b>	‘C’: 1, ‘degree’: 2 ‘gamma’: 0.01	<b>0.820</b>
Sigmoid	‘C’: 10, ‘gamma’: 0.001	0.817

The bold values signifies the higher results found in the comparison, for all splits.

**Table 7**

Overall comparison between CNN-MLP, CNN-SVM, Xception, ResNet50, InceptionV3, and the method proposed in [5], on 4-class classification with 90/10 split.

Method	$\mu P$	$\mu R$	$\mu F1$	$\mu Acc$
Xception	0.786(±0.050)	0.782(±0.050)	0.775(±0.051)	0.782(±0.050)
ResNet50	0.791(±0.025)	0.782(±0.029)	0.776(±0.028)	0.782(±0.029)
InceptionV3	0.788(±0.037)	0.775(±0.036)	0.772(±0.035)	0.775(±0.036)
CNN-MLP	0.750(±0.041)	0.752(±0.033)	0.740(±0.037)	0.752(±0.033)
CNN-SVM	<b>0.829(±0.038)</b>	<b>0.820(±0.040)</b>	<b>0.819(±0.039)</b>	<b>0.820(±0.040)</b>
[5]	0.518(±0.052)	0.539(±0.004)	0.528(±0.007)	0.539(±0.004)

The bold values signifies the higher results found in the comparison, for all splits.

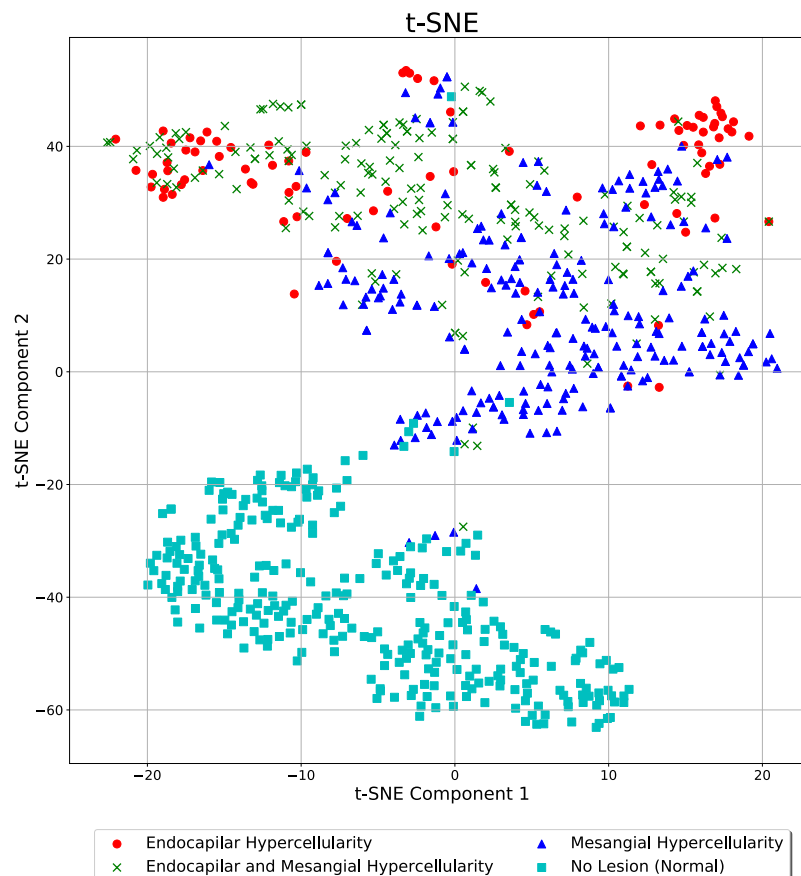
**Table 8**

Number of parameters of the evaluated CNN models.

Model	# parameters
Xception	22,910,480
ResNet50	25,636,712
InceptionV3	23,851,784
CNN-MLP	1,220,128
CNN-SVM (backbone)	431,904

binary classification methodology was followed, but now maintaining  $K=10$  on the cross-validation. As expected, the only modification on the CNN architecture was the number of dense layers at the top of the model, since the number of classes was changed. At each fold on cross-validation, weights from the best CNN-MLP model on binary classification were loaded, updating only the number of classes on the last layer. Then, the whole CNN-MLP model was retrained on the 4-class data set using the same former training parameters, achieving an average accuracy of 75.2%. Just as the binary classification, the best

model was selected among the 10 models from each fold of cross-validation, using the CNN backbone as a feature extractor, feeding an SVM classifier. The kernel parameters were varied in the same way as in the former experiments, achieving, as the best result, an average accuracy of 82%. Table 6 shows the parameters of the best results for each SVM kernel; while Table 7 summarizes the final results for comparison between CNN-MLP and CNN-SVM along with the other three deep models and the method proposed by Barros et al. [5]. Even though the polynomial kernel had achieved the best results, all kernels had similar scores, proving the robustness of the CNN architecture for feature extraction. Since Barros et al. [5] developed a handcrafted feature extractor for binary classification, it is expected a lower performance on 4-class classification. This underperformance did not occur when using deep learning models, which, as explained in Section 2.1, do not depend on the domain problem. Although the CNN-MLP model has not surpassed the three deep models, the proposed model achieved a very close performance. This behavior may indicate that the 4-class classification could need a deeper architecture to achieve better accuracy on a full end-to-end network. Regardless, the CNN-SVM outperformed all the



**Fig. 5.** *t*-SNE visualization of the 4-class data set. The CNN feature extractor generates a 128-dimensional feature vector, and the *t*-SNE algorithm reduces the dimensionality to a 2-dimensional vector to help the analysis of the clusters.



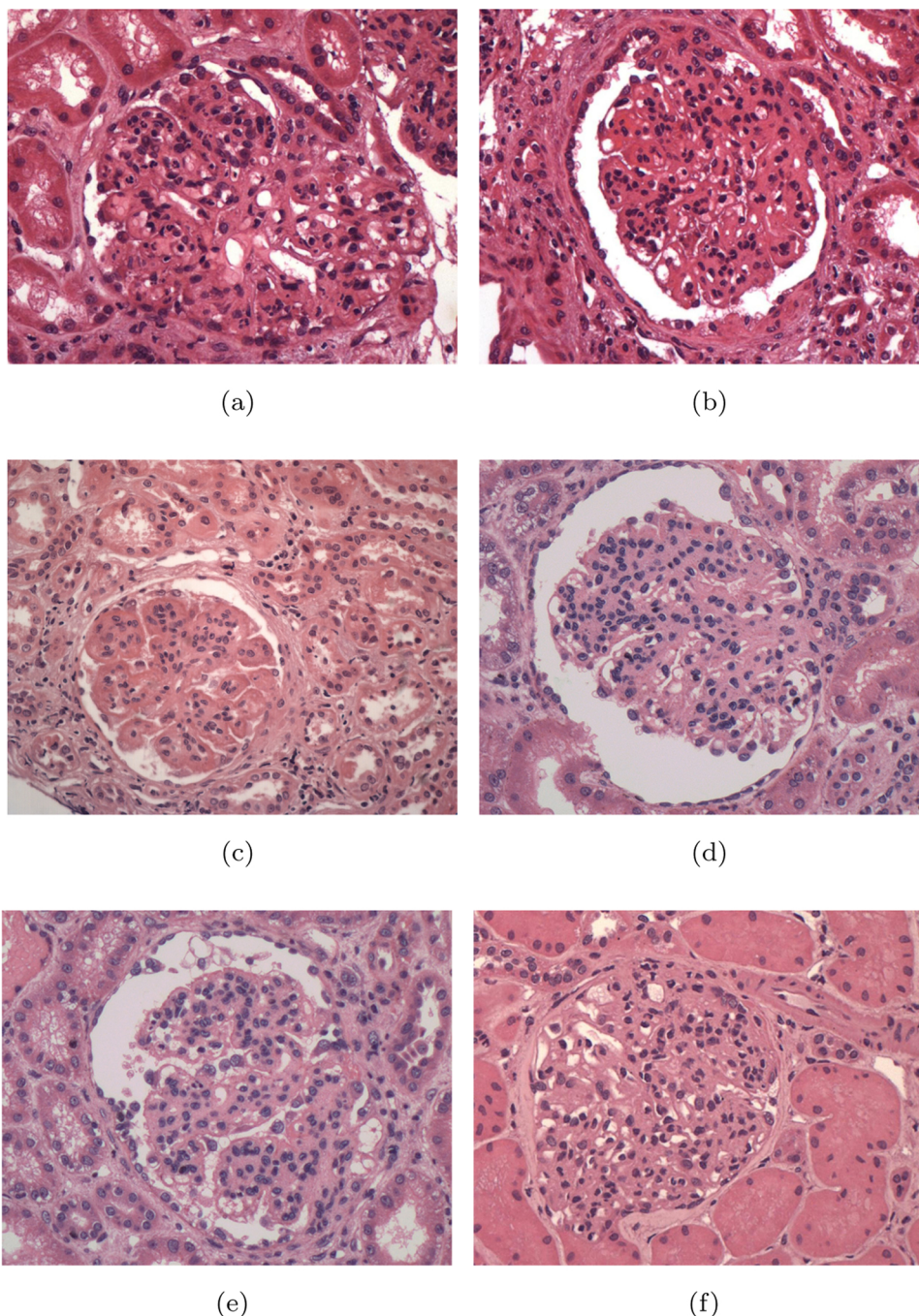
**Table 9**  
Confusion Matrix sum over the 10-fold cross-validation using CNN-SVM model. The rows represent the predicted classes; the columns represent the true classes.

	Endocapillary	<i>EndoMes</i>	Mesangial	Normal
Endocapillary	53	24	13	0
<i>EndoMes</i>	28	122	29	0
Mesangial	9	31	198	0
Normal	0	1	11	292

other models, proving the robustness of the feature extractor combined with the SVM classifier.

#### 4. Discussion and conclusions

On binary classification, the two classification approaches (CNN-MLP and CNN-SVM) achieved high overall results on all metrics with low standard deviations, as showed in Tables 1 and 4. The two methods had close results, with CNN-SVM approach showing a slightly better performance for every value of *K*, proving the robustness of the final proposed model. Despite the unbalanced data set (more samples for



**Fig. 6.** Six images of misclassified glomeruli with CNN-SVM architecture. From the left to right: (a) endocapillary hypercellularity misclassified as mesangial hypercellularity, (b) endocapillary hypercellularity misclassified as *endoMes* hypercellularity, (c) mesangial hypercellularity misclassified as endocapillary hypercellularity, (d) mesangial hypercellularity misclassified as *endoMes* hypercellularity, (e) *endoMes* hypercellularity misclassified as endocapillary hypercellularity, and (f) *endoMes* hypercellularity misclassified as mesangial hypercellularity.

**Table 10**  
Comparison between the pathologists' labels and the results obtained by the trained CNN-SVM model. The *Pool* column represents the majority vote: computer (COMP) or pathologist (PAT).

Image (see Fig. 6)	Classifier				Pool
	Pathologist 1	Pathologist 2	Pathologist 3	CNN-SVM	
a	END	END	END	MES	PAT
b	END	ENDOMES	ENDOMES	ENDOMES	COMP
c	MES	END	ENDOMES	END	COMP
d	MES	MES	MES	ENDOMES	PAT
e	ENDOMES	MES	ENDOMES	END	PAT
f	ENDOMES	ENDOMES	END	MES	PAT

lesion than for normal glomeruli), we did not observe the models being heavily biased on the class with more images. This behavior may be due to two factors: Image augmentation and feature quality. The process of image augmentation helped to solve this issue by increasing the number of images through random modifications on the original training set. The features obtained from the CNN backbone proved to be highly suitable for classification using all kernels, achieving an average accuracy of 100% on the linear kernel. This outcome demonstrates that, despite the size of the CNN features (128), these features are linearly separable, which is an outstanding finding.

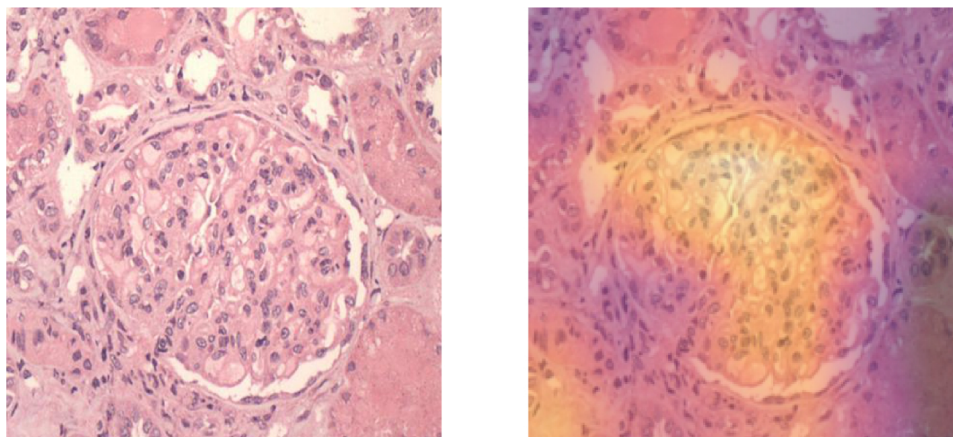
A summary of the results of binary classification is presented in Table 5, displaying the best results of the CNN-MLP and CNN-SVM models in comparison with other three deep models and the method proposed by Barros et al. [5] for all splits. To the best of our knowledge, Barros et al. [5] were the first to address the problem of glomerular hypercellularity lesion classification so far, what demonstrates that we achieved an improvement of 12 percentage points in accuracy with our proposed deep learning-based model on the same data set. The proposed models also outperformed the other state-of-the-art methods with much fewer parameters, as summarized in Table 8. As the main outcome, we obtained perfect accuracy with at least 53 times fewer parameters than the other state-of-the-art deep-learning architectures, and achieved a linear feature separation.

Considering the 4-class classification, both CNN-MLP and CNN-SVM models achieved high results, even though the gap between these two approaches has increased (approximately seven percentage points) for all metrics, as summarized in Table 7. This behavior may have occurred due to the difficulty of differentiating the 4 classes, mainly with respect to the sub-lesions. Another relevant characteristic is the *endoMes* class, which contains features that can be confused with both endocapillary and mesangial hypercellularity. Fig. 5 illustrates the feature space of the data set plotted using the *t*-distributed stochastic neighbor embedding

(*t*-SNE), which is a common technique for visualizing high-dimensional data into 2-dimensional plots. It is noteworthy that the “no lesion” class is well separated from the other lesion classes, which explains the 100% accuracy of the binary classification. The three lesion classes have some well-defined groups, but these classes also have some areas with quite an overlap of instances, meaning that images containing endocapillary, mesangial and *endoMes* hypercellularity can be very similar. Overall, the endocapillary class is well separated from the mesangial one, but the *endoMes* class overlaps between the two groups. To confirm these findings, we summed the confusion matrices over the 10 folds on cross-validation of the CNN-SVM best model, and results are shown in Table 9. One finding was that there is no false negative, indicating a good performance on differentiating lesion from non-lesion (performance already identified on binary classification). The zero false negative rate is especially important in diagnosis tasks, avoiding that lesioned glomeruli to be misclassified as normal ones.

Fig. 6 shows six images misclassified by the CNN-SVM model, considering every possible error combination. These images depict complex lesions that may represent a challenge even for nephropathologists (corroborating with the *t*-SNE visualization). Fig. 6(a) represents a glomeruli with increased circularity caused by cell proliferation and influx of inflammatory cell with disruption of glomerular compartments. Fig. 6(b) represents a glomeruli with hypercellularity combined with mesangial matrix expansion and capillary wall thickening probably by immune complex deposition on the subendothelial and on the subepithelial aspects of the glomerular basement membrane burling the limits of glomerular compartments. Fig. 6(c) hypercellularity is combined with capillary wall thickening and partial mesangial dissolution. In Figs. 6(d) and (f), mesangial and capillary lumen are not always well defined. We showed these six images to be independently classified by three pathologists. The results of this analysis are shown in Table 10. Complete agreement among nephropathologists on the distribution of hypercellularity was achieved only in two out of the six images. In diagnostic practice most of the difficulties generated by these complex lesions are usually solved by examining contiguous tissue sections of 2-10 μm apart, stained with a variety of techniques to highlight basement membrane and mesangial matrix such as PAS and periodic acid-methenamine silver (PAMS).

Even though we have compared pathologist and model predictions in the consensus analysis, the pathologist and machine classification processes are very different. Barros et al. [5] used nuclei segmentation and estimation of the number of clusters for feature extraction, resulting in an understandable set of features for doctors. Our proposed model (and deep learning models in general) performs feature extraction using several convolutions and matrix operations, which cannot return a “human-readable” set of features. However, several



**Fig. 7.** Left image: glomerulus with endocapillary lesion. Right image: GRAD-CAM heatmap [33] visualization of relevant regions for the 4-class CNN-MLP model. The hotter the glomerulus area, the more important is the region to the CNN as a feature.

visualization techniques have been proposed to interpret deep learning model predictions, where one of these techniques is the class activation map (CAM) [43]. Aiming to illustrate a form of visualization of the model prediction, we used Grad-CAM [33], which is a CAM technique variation, to “visually explain” the decision of the classifier. Fig. 7 shows a glomerulus with an endocapillary lesion (correctly classified by the 4-class CNN-MLP model), and in the right, the Grad-CAM visualization highlighting the important regions for prediction by the CNN-MLP model. The visualization is a heatmap plotted over the original image, where the “hottest” regions are more important to the classification. One cannot state that the regions relevant to classification are also pathologically relevant, so we leave this field to future research.

For future work, we are investigating different ways of using a transfer learning approach to initialize our network with better weights for generalizing glomerulus image classes, where sufficient training data exists. Additionally, we plan to expand the number of samples (now around 31,000 unlabeled images) in the data set, working with other types of lesions and histological stains used in the pathology laboratory for better data analysis. Another work in progress is the automatic glomerulus segmentation in a WSI containing several glomeruli; the goal is to classify each found segmented glomerulus.

## 5. Ethical considerations

This work was conducted in accordance with resolution No. 466/12 of the Brazilian National Health Council. To preserve confidentiality, the images (including those shown in the paper) were separated from other patient's data. No data presented herein allows patient identification. All the procedures were approved by the Ethics Committee for Research Involving Human subjects of the Gonçalo Moniz Institute from the Oswaldo Cruz Foundation (CPqGM/FIOCRUZ), Protocols No. 188/09 and No. 1817574.

## Conflict of interest

None declared.

## Acknowledgment

Authors are grateful to Fundação de Amparo à Pesquisa do Estado da Bahia (FAPESB) grants TO-SUS0031/2018 and TO-PET0008/2015. Luiz Souza and Paulo Chagas have scholarships from FAPESB, grants TO-BOL0660/2018 and TO-BOL0344/2018, respectively. Washington Santos and Luciano Oliveira have research scholarships from Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), grants 306779/2017 and 307550/2018-4, respectively.

## References

- Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: large-scale machine learning on heterogeneous systems. 2015 Software available from <https://www.tensorflow.org/>.
- Al-Janabi S, Huisman A, Van Diest PJ. Digital pathology: current status and future perspectives. *Histopathology* 2012;61(1):1–9.
- Bajema IM, Wilhelmus S, Alpers CE, Bruijn JA, Colvin RB, Cook HT, et al. Revision of the international society of nephrology/renal pathology society classification for lupus nephritis: clarification of definitions, and modified national institutes of health activity and chronicity indices. *Kidney Int* 2018;93(4):789–96.
- Barisoni L, Nast CC, Jennette JC, Hodgin JB, Herzenberg AM, Lemley KV, et al. Digital pathology evaluation in the multicenter nephrotic syndrome study network (neptune). *Clin J Am Soc Nephrol* 2013;8(8):1449–59.
- Barros GO, Navarro B, Duarte A, Dos-Santos WL. Pathospotter-k: a computational tool for the automatic identification of glomerular lesions in histological images of kidneys. *Sci Rep* 2017;7:46769.
- Canziani A, Paszke A, Cullurciello E. An analysis of deep neural network models for practical applications. 2016arXiv:1605.07678.
- Chapelle O, Vapnik V, Bousquet O, Mukherjee S. Choosing multiple parameters for support vector machines. *Mach Learn* 2002;46(1–3):131–59.
- Chollet F. Xception: deep learning with depthwise separable convolutions. 2016arXiv:1610.02357.
- Churg J, Bernstein J, Glassock R. Renal disease: classification and atlas of glomerular diseases. New York, Tokyo: Igaku-Shoin; 1995.
- Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. Imagenet: a large-scale hierarchical image database. 2009 IEEE conference on computer vision and pattern recognition 2009:248–55.
- Duan K, Keerthi SS, Poo AN. Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing* 2003;41–59.
- Fabjanska A. Segmentation of corneal endothelium images using a u-net-based convolutional neural network. *Artif Intell Med* 2018;88:1–13.
- Fogo AB. Approach to renal biopsy. *Am J Kidney Dis* 2003;42(4):826–36.
- Fu X, Ong C, Keerthi S, Hung GG, Goh L. Extracting the knowledge embedded in support vector machines. *IEEE international joint conference on neural networks*, vol. 1 2004:291–6.
- Gandomkar Z, Brennan PC, Mello-Thoms C. Modern: multi-category classification of breast histopathological image using deep residual networks. *Artif Intell Med* 2018;88:14–24.
- Gu J, Wang Z, Kuen J, Ma L, Shahroudy A, Shuai B, et al. Recent advances in convolutional neural networks. 2015arXiv:1512.07108.
- He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. 2015arXiv:1512.03385.
- He T, Zhang Z, Zhang H, Zhang Z, Xie J, Li M. Bag of tricks for image classification with convolutional neural networks. *Proceedings of the IEEE conference on computer vision and pattern recognition* 2019:558–67.
- Hou L, Samaras D, Kurc TM, Gao Y, Davis JE, Saltz JH. Patch-based convolutional neural network for whole slide tissue image classification. *IEEE conference on computer vision and pattern recognition* 2016:2424–33.
- Imbault F, Lebart K. A stochastic optimization approach for parameter tuning of support vector machines. *International conference on pattern recognition*, vol. 4 2004:597–600.
- Janowczyk A, Madabhushi A. Deep learning for digital pathology image analysis: a comprehensive tutorial with selected use cases. *J Pathol Informatics* 2016:7.
- Joosten SA, Sijpkens YW, van Kooten C, Paul LC. Chronic renal allograft rejection: pathophysiologic considerations. *Kidney Int* 2005;68(1):1–13.
- Kannan S, Morgan LA, Liang B, Cheung MG, Lin CQ, Mun D, et al. Segmentation of glomeruli within trichrome images using deep learning. *Kidney Int Rep* 2019.
- Kingma DP, Ba J. Adam: a method for stochastic optimization. 2014arXiv:1412.6980.
- LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature* 2015;521(7553):436–44.
- Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, et al. A survey on deep learning in medical image analysis. *Med Image Anal* 2017;42:60–88.
- Madabhushi A, Lee G. Image analysis and machine learning in digital pathology: challenges and opportunities. *Med Image Anal* 2016;33:170–5.
- Marsh JN, Matlock MK, Kudose S, Liu TC, Stappenbeck TS, Gaut JP, et al. Deep learning global glomerulosclerosis in transplant kidney frozen sections. *bioRxiv* 2018:292789.
- Pedraza A, Gallego J, Lopez S, Gonzalez L, Laurinavicius A, Bueno G. Glomerulus classification with convolutional neural networks. *Annual conference on medical image understanding and analysis* 2017:839–49.
- Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: machine learning in Python. *J Mach Learn Res* 2011;12:2825–30.
- Sarder P, Ginley B, Tomaszewski JE. Automated renal histopathology: digital extraction and quantification of renal pathology. *Medical imaging 2016: digital pathology*, vol. 9791 2016:97910F.
- Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-cam: visual explanations from deep networks via gradient-based localization. 2016arXiv:1610.02391.
- Sharma H, Zerbe N, Klempert I, Hellwich O, Hufnagl P. Deep convolutional neural networks for automatic classification of gastric carcinoma using whole slide images in digital histopathology. *Comput Med Imaging Graph* 2017;61:2–13.
- Simon O, Yacoub R, Jain S, Tomaszewski JE, Sarder P. Multi-radial LBP features as a tool for rapid glomerular detection and assessment in whole slide histopathology images. *Sci Rep* 2018;8(1):2032.
- de Souza BF, de Carvalho AC, Calvo R, Ishii RP. Multiclass SVM model selection using particle swarm optimization. *International conference on hybrid intelligent systems* 2006:31.
- Spanhol FA, Oliveira LS, Petitjean C, Heutte L. Breast cancer histopathological image classification using convolutional neural networks. *International joint conference on neural networks* 2016:2560–7.
- Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. 2015arXiv:1512.00567.
- Trimarchi H, Barratt J, Cattran DC, Cook HT, Coppo R, Haas M, et al. Oxford classification of IGA nephropathy 2016: an update from the IGA nephropathy classification working group. *Kidney Int* 2017;91(5):1014–21.
- Wahab N, Khan A, Lee YS. Two-phase deep convolutional neural network for reducing class skewness in histopathological images based breast cancer detection. *Comput Biol Med* 2017;85:86–97.
- Xu J, Luo X, Wang G, Gilmore H, Madabhushi A. A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images. *Neurocomputing* 2016;191:214–23.
- Zhang G, Hsu CHR, Lai H, Zheng X. Deep learning based feature representation for automated skin histopathological image annotation. *Multimedia Tools Appl* 2018;77(8):9849–69.
- Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features 605 for discriminative localization. 2015arXiv:1512.04150.