Journal of Pathology Informatics 13 (2022) 100093

Contents lists available at ScienceDirect

Journal of Pathology Informatics

journal homepage: www.elsevier.com/locate/jpi

Original Research Article

A deep learning framework for automated classification of histopathological kidney whole-slide images

Hisham A. Abdeltawab [a], Fahmi A. Khalifa [a], Mohammed A. Ghazal [a], Liang Cheng [b,c], Ayman S. El-Baz [a], Dibson D. Gondim [b,*]

[a] Bioengineering Department, University of Louisville, Louisville, KY, USA
[b] Department of Pathology and Laboratory Medicine, University of Louisville School of Medicine, Louisville, KY, USA
[c] Department of Pathology and Laboratory Medicine, Department of Urology, Indiana University School of Medicine, Indianapolis, IN, USA

ARTICLE INFO

Keywords:
Histopathological images
Computational pathology
Kidney cancer
Deep learning

ABSTRACT

Background: Renal cell carcinoma is the most common type of malignant kidney tumor and is responsible for 14,830 deaths per year in the United States. Among the four most common subtypes of renal cell carcinoma, clear cell renal cell carcinoma has the worst prognosis and clear cell papillary renal cell carcinoma appears to have no malignant potential. Distinction between these two subtypes can be difficult due to morphologic overlap on examination of histopathological preparation stained with hematoxylin and eosin. Ancillary techniques, such as immunohistochemistry, can be helpful, but they are not universally available. We propose and evaluate a new deep learning framework for tumor classification tasks to distinguish clear cell renal cell carcinoma from papillary renal cell carcinoma.
Methods: Our deep learning framework is composed of three convolutional neural networks. We divided whole-slide kidney images into patches with three different sizes where each network processes a specific patch size. Our framework provides patchwise and pixelwise classification. The histopathological kidney data is composed of 64 image slides that belong to 4 categories: fat, parenchyma, clear cell renal cell carcinoma, and clear cell papillary renal cell carcinoma. The final output of our framework is an image map where each pixel is classified into one class. To maintain consistency, we processed the map with Gauss-Markov random field smoothing.
Results: Our framework succeeded in classifying the four classes and showed superior performance compared to well-established state-of-the-art methods (pixel accuracy: 0.89 ResNet18, 0.92 proposed).
Conclusions: Deep learning techniques have a significant potential for cancer diagnosis.

1. Introduction

Kidney cancer is among the 15 most common cancers in humans and is responsible for about 2.4% of cancers.[1] The American Cancer Society estimates 73,750 new patients will be diagnosed with kidney cancer in the United States in 2020 and approximately 14,830 people will die from the disease.[2] Renal cell carcinoma (RCC) is the most prevalent type and accounts for about 85% of kidney cancers. RCC is a heterogeneous group of cancers with different molecular characteristics, histology, responses to therapy, and clinical outcomes.[3] The most common subtypes of RCC are clear cell (70–80%), papillary (14–17%), chromophobe (4–8%), and clear cell papillary RCC (4%).[4] Classification of subtypes of RCC is based primarily on the morphologic features observed on histological preparation stained with hematoxylin and eosin. Among the four most common subtypes of RCC, clear cell RCC and clear cell papillary RCC have significant morphologic overlap, notably the frequent presence of clear cells. Distinction of clear cell RCC and clear cell papillary RCC is critical to determine appropriate patient management. Clear cell RCC can have a poor prognosis due to high risk of metastatic spread. This contrasts with clear cell papillary RCC, which is recognized as an indolent neoplasm that has not be proven to metastasize.[5] In cases with significant morphologic overlap, immunohistochemistry can be a helpful tool to separate these two types of tumors. However, immunohistochemistry is not routinely available in all parts of the world. Deep learning algorithms achieved excellent results in various histopathology tasks including for tumor segmentation, grading, and classifications task.[6–11] Regarding classification of RCC, a classifier based on Resnet successfully distinguished the three most common subtypes of RCC.[12] However, it was not trained to separate clear cell RCC from clear cell papillary RCC. The purpose of this work was to create a deep learning classifier to distinguish clear cell RCC from clear cell papillary RCC which can be a difficult task and clinically relevant.

We propose a computerized-aided diagnostic (CAD) system that is based on convolutional neural network (CNN) for the automated classification that addresses this differential diagnosis. Our framework has the following contributions:

* Corresponding author at: Department of Pathology and Laboratory Medicine, University of Louisville School of Medicine, 530 S. Jackson St, Louisville, KY, USA.
E-mail address: dibson.gondim@louisville.edu (D.D. Gondim).

http://dx.doi.org/10.1016/j.jpi.2022.100093

Available online 18 April 2022

- An ensemble-pyramidal deep learning model that utilizes a hierarchy of three CNN that process different image sizes.
- Provides both patchwise classification and pixelwise classification.
- Incorporates a statistical approach based on Markov-Gibbs random field to improve the final pixelwise classification.

2. Methods

This study was approved by the institutional review board. All hematoxylin and eosin-stained slides collected for the study were deidentified before digitalization. The image files used have no identifiers that link to a patient.

Cases of clear cell RCC (27) and clear cell papillary RCC (14) were randomly selected from the institution files. The cases were reviewed by a pathologist with expertise in genitourinary pathology (D.D.G). One representative slide was selected from each case and scanned using Phillips UFS at 40× magnification. The images were downsampled by a factor of 4 and downloaded. The images were segmented by a pathologist into four groups: clear cell RCC, clear cell papillary RCC, renal parenchyma, and fat. The following subsections describes our methodology and for more details please refer to Abdeltawab et al.[13] To assess the generalizability of our method, another dataset from an external source (NIH)[14] has been used to validate our methodology. The external dataset is composed of 10 whole slide images of clear cell RCC.

2.1. Patches generation

We divided the data (from our institution) into two sets. The first dataset, for training and testing, is composed of 44 image slides. The first set was then divided into two parts to perform training using the first part and testing using the second part. This testing is considered a first inference using unseen data to evaluate the patch-wise accuracy. The second dataset, for final validation, is composed of 10 image slides (2 fat, 2 parenchyma, 3 clear cell papillary RCC, and 3 clear cell RCC). For the first dataset, each slide image was divided into overlapping patches of three different sizes: small size = 250 × 250, medium size = 350 × 350, and large size = 450 × 450. The use of different patch sizes allows the recognition of features at various scales. Learning features at different scales by using the three CNNs resulted in better performance than learning features at a single scale by using a single CNN. The degree of overlap between each patch and the next one was 50%. This will let the deep learning model learn various viewpoints within the tissue. Then, the patches that contained only background pixels were removed. About 70% of the patches was kept for training and about 30% of the patches was kept for testing. Similarly, for the second dataset (i.e., validation set), the slide images were divided into patches of the three sizes with only a 5-pixel shift in both image dimensions between each patch. Therefore, we increased the degree of overlap in the second dataset. This will allow us to have multiple labels for the same region during testing. The second dataset will be used for final validation and pixelwise classification. This validation is considered a second inference where we evaluated the patch- and pixel-wise accuracies. The slides of the external dataset were divided into overlapping patches of the three mentioned sizes. Fig. 1 shows samples for each tissue type at different patch sizes.

2.2. Preprocessing

We applied two preprocessing steps on the patches of the first and second datasets: an adaptive histogram equalization, and edge enhancement.

2.2.1. Adaptive histogram equalization

Adaptive histogram equalization is an image processing approach used to enhance the contrast of the image.[15] Using this approach, we compute several histograms, each histogram corresponding to a different part in the image. Ordinary histogram equalization utilizes only one

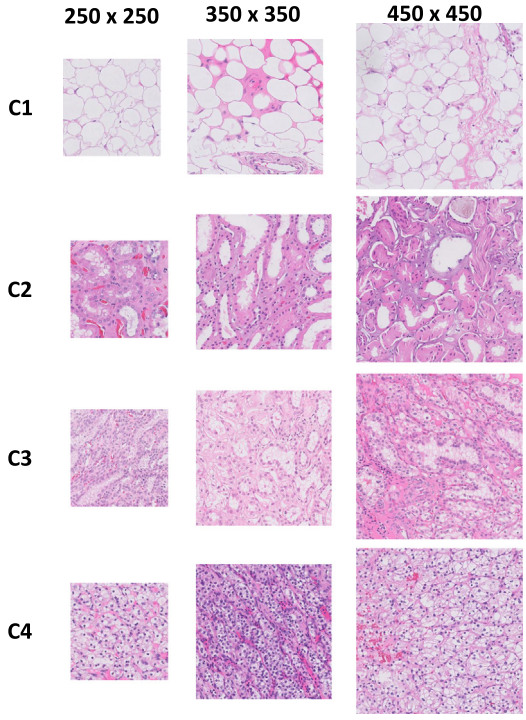

Fig. 1. Samples for each tissue type where C1, C2, C3, and C4 refer to fat, parenchyma, clear cell papillary RCC, and clear cell RCC.

histogram for the whole image. Adaptive histogram equalization uses the computed histograms to redistribute the intensity values of the image. Therefore, it can be used to enhance the local contrast and improve the edges in each part of an image. Because the homogenous parts of the image have highly concentrated histograms, adaptive histogram equalization tends to overamplify the noise. To prevent this, we use a variant called contrast limited adaptive histogram equalization which avoids amplifying the noise by putting a limit on the amplification. In contrast limited adaptive histogram equalization, the equalization step is followed by interpolating neighboring regions using bilinear interpolation to remove artificially generated boundaries.

2.2.2. Image edge enhancement

We applied edge enhancement to all patches of the training and testing set. In image edge enhancement, we work on enhancing the edge of important objects in the image. The purpose of image edge enhancement is to improve the visual perception of the image. The enhancement is performed by suppressing the low frequencies of the image, which is called image filtering. The filtering can be performed in the spatial or the frequency domain. We perform filtering by convolving the image in the spatial domain with a sharpening filter. Suppose an input patch is x(m,n) and the used filter (Fig. 2) is h(m,n), then the filtered patch y(m,n) is given by:

$$y(m,n) = x(m,n)^{*}h(m,n) \quad (1)$$

2.3. The proposed deep learning architecture

Following a proposed deep learning method schema (Fig. 3), we use the extracted patches to train three CNN with three input sizes: 250 × 250, 350 × 350, and 450 × 450. We refer to these CNN by CNN_S, CNN_M, CNN_L for small, medium, and large patch size, respectively. The three CNN shared the same architecture. As illustrated in Fig. 3, the architecture of the CNN models is represented by a series of convolution layers intervened by max-pooling layers followed by two fully connected layers. Finally, there is a soft-max layer. In the convolution layer, the image is convolved with

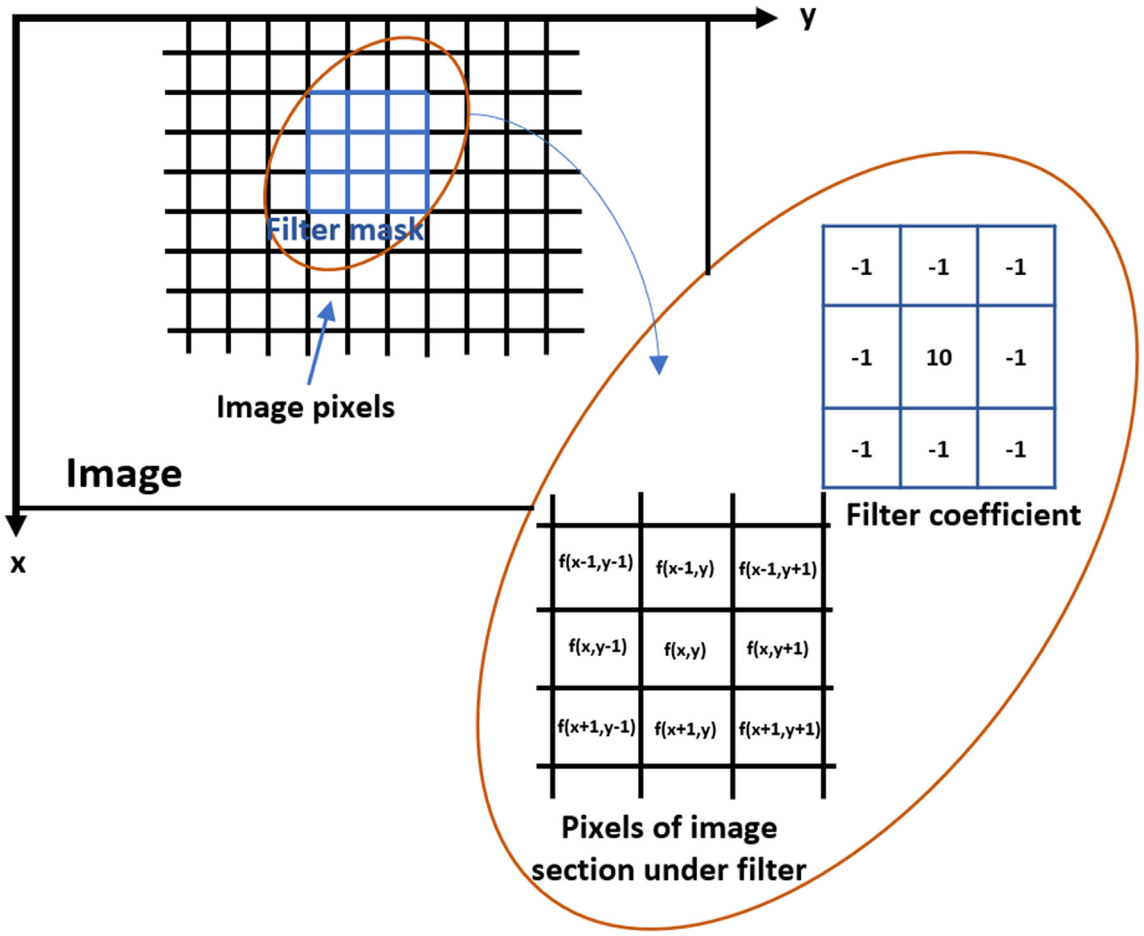

Fig. 2. The filter/kernel used for patch edge enhancement.

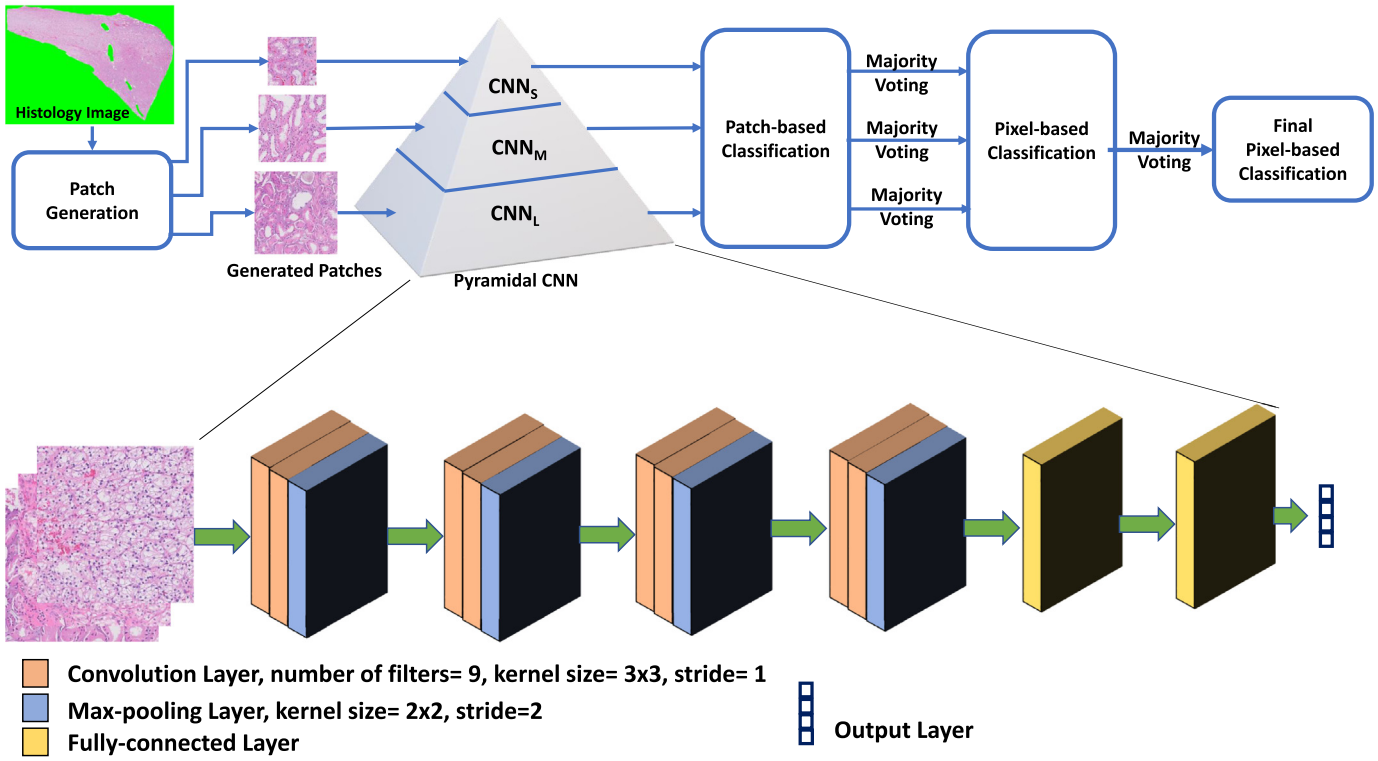

Fig. 3. The proposed deep learning model for the classification of kidney histopathological images.

kernels to extract features that describe the objects in the input image; these features are called feature maps. There are multiple kernels in each layer. Therefore, each convolution layer resulted in a volume of feature maps. In our implementation, we used kernels of size 3 × 3 that had stride = 1. In max-pooling layers, the spatial dimensions were reduced by a factor of 2. This kept only the most important features and discarded those less prominent. Also, the max-pooling operation reduced the computational cost and training time. In our implementation, the stride of the max-pooling layers was 2. Each CNN contained eight convolution layers and four max-pooling layers. The first fully connected layer contained 12 neurons and the second one contained 4 neurons because we had four classes. The final soft-max layer was responsible for the multiclass classification because it took the input number from the previous fully connected layer and produced values in the range of 0–1. Therefore, the soft-max layer assigned a probability for each class. The highest probability indicated the final classification result for that input image. The implemented CNN were trained by minimizing the cross-entropy loss. In the loss function, we minimized the cross entropy between the predicted probabilities and the ground truth labels. To minimize network overfitting, a dropout with a rate of 0.2 was used in the convolution and fully connected layers.

Each CNN had a patchwise accuracy and a pixelwise accuracy for a tested slide image. It had a patchwise accuracy because it can classify the input patch to one of the four classes. The patchwise accuracy for one slide image was estimated as follows:

$$Accuracy_{patch} = \frac{N_{cp}}{N_{tp}} \tag{2}$$

Where N_{cp} is the number of correctly classified patches, and N_{tp} is the total number of patches. Then, we estimated the average of accuracies for all slide images. In the second dataset, we had a 5-pixel shift between each patch. Therefore, each pixel in the image possibly belonged to many patches. During classification of a single patch, we assigned the label of the patch to its pixels. Therefore, each pixel could have multiple labels from multiple patches. Majority voting was applied for the labels of the pixels to obtain one label for each pixel, and we constructed a labeled slide image for each slide image. The pixelwise accuracy for a single slide image was estimated as follows:

$$Accuracy_{pixel} = \frac{N_{cx}}{N_{tx}} \tag{3}$$

Where N_{cx} is number of correctly classified pixels, and N_{tx} is the total number of pixels. Then, we estimated the average of accuracies for all slide images.

To obtain a better pixelwise accuracy, the results of the three CNN were combined. Because we have three CNN, we had three labels for each pixel in a tested slide image. Again, we applied majority voting to get the final label for each pixel. Finally, we estimated the final pixelwise accuracy and the average over all images, as previously discussed.

2.4. Gauss-Markov random field smoothing

During patch-based classification, the label of the patch is assigned to the pixels of the patch. A pixel can have multiple labels from multiple patches. The result of the majority voting is to assign a single label for each pixel. Then, a labeled slide image is constructed for the whole slide image. We removed inconsistencies (smooth) and preserved continuity in the labeled image by considering the estimated labels γ samples from a generalized 2-D Gauss-Markov random field (GGMRF) model.[16] Each pixel had an 8-neighborhood (Fig. 4). Continuity of the γ values was amplified using their voxelwise stochastic relaxation[17] and maximum a posteriori estimates:

$$\widehat{\gamma}_O = arg_{\widehat{\gamma}_O} \min \left\{ |\gamma_O - \widehat{\gamma}_O|^{\theta} + \rho^{\theta}\mu^{\varphi} \sum_{r \in \upsilon_O} \sigma_{O,r} |\widehat{\gamma}_O - \gamma_r|^{\varphi} \right\} \tag{4}$$

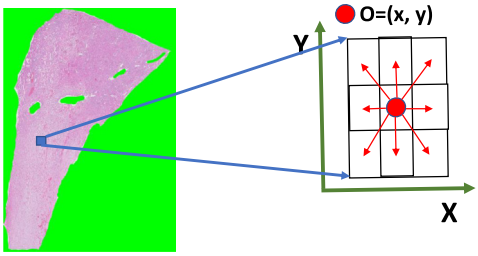

Fig. 4. Pairwise voxel interaction for the 8 neighbors in a 2-D GGMRF image model for continuity analysis.

Where γ_O and $\widehat{\gamma}_O$ refer to the original label and its expected estimates, respectively, at the observed 2-D location, O = (x,y), $\sigma_{O,r}$ is the GGMRF potential, υ_O is the 8-neighborhood voxel set (Fig. 4), and ρ and μ are scaling factors. The parameter $\theta \in \{1, 2\}$ determines the Laplace, $\theta = 1$ or the Gaussian, $\theta = 2$, prior distribution of the estimator. The parameter $\varphi \in [1.01,2.0]$ controls the level of smoothing (e.g., $\varphi = 1.01$ for relatively abrupt edges versus $\varphi = 2$ for smooth edges). In our calculations, we set $\varphi = 1.01, \theta = 2, \rho = 1, \mu = 5$, and $\sigma_{O,r} = \sqrt{2}$. We hypothesized that applying smoothing for the labeled slide image would increase the pixelwise accuracy.

3. Results

Our deep learning framework was developed using TensorFlow library from Google.[18] To determine the best parameters of our framework, such as network architecture, we analyzed system performance as a function of the important parameters. Therefore, we performed a grid search to determine the best settings for the number of convolutional layers (range 2:2:10), kernel size (values 3 × 3, 5 × 5, and 7 × 7), number of filters (range 3:3:12), initialization of the convolutional layers (Glorot Uniform distribution or He initialization), stride (range 1:2), number of epochs (range 50:10:80), patch size (values 16 or 32), type of optimizer (Adam or stochastic gradient descent), and learning rate (values 0.01, or 0.001). The result of the grid search was as follows: number of convolutional layers was equal to 8, kernel size was equal to 3 × 3, number of filters was equal to 9, the initialization of convolutional layers was set to He initialization,[19] stride of the convolution was set to 1 and stride of the max-pooling was set to 2, number of epochs was set to 60, patch size was set to 32, and we used Adam optimizer with 0.001 learning rate. Twenty percent of the training data was kept for validating the network after each epoch by estimating the validation loss and the validation accuracy. The final neural network was chosen based on the value of the validation accuracy. To increase the size of the training data, we used data augmentation in the form of random rotation, flipping, and scaling. Differently scaling objects of interest in the images is the most important aspect of image diversity. Therefore, this data augmentation technique increases the prediction accuracy. Our scaling technique does not result in significant blur as it produces images with a random scaling factor that ranges from 100% to 80% of the original image.

We trained and tested our framework using the respective sets of the first dataset. During training and testing, we fed each patch size to its respective CNN. After testing, we estimated the patchwise accuracy for each tissue type. Table 1 presents the patchwise accuracy for the four tissue types.

To assess the performance of our proposed framework, we compared it with pretrained ResNet18 and ResNet34. In the ResNet networks, we replaced the last layers with two output layers. Then, we fine-tuned them on the kidney data. Table 6 shows the average pixelwise accuracy of our approach versus ResNets. To find a significant difference between our method and other deep learning models, we performed paired one-tailed *t*-tests with the assumptions that the data are continuous (not discrete) and the differences for the matched-pairs, follow a normal probability distribution.

Table 1
Patchwise accuracy for each tissue type using the testing set at different patch sizes.

	Patch size		
	250 × 250	350 × 350	450 × 450
Fat	0.89 ± 0.01	0.91 ± 0.02	0.90 ± 0.11
Parenchyma	0.88 ± 0.11	0.91 ± 0.12	0.90 ± 0.24
Clear cell papillary RCC	0.86 ± 0.23	0.90 ± 0.21	0.89 ± 0.21
Clear cell RCC	0.85 ± 0.02	0.89 ± 0.31	0.89 ± 0.03

The *P* value of the test was 0.01 for Resnet18 and 0.02 for Resnet34. The *P* values were less than 0.05. Therefore, we rejected the null hypothesis in the two cases and concluded that there was significant difference between our proposed method and the other deep learning models.

In this paper, we proposed an automatic CAD system that can classify kidney histopathological images. Specifically, we target the hard task of classifying between clear cell RCC and clear cell papillary RCC. These cancer subtypes share morphological features but have different prognoses. Therefore, our task is important to determine the correct patient management. We utilized the power of deep learning to build our CAD system. We developed a novel ensemble-pyramidal deep learning model that is based on three CNN. Our pyramidal-CNN successfully identified the normal and abnormal tissue textures and classified clear cell RCC vs clear cell papillary RCC. We also introduced the notion of pixelwise accuracy using a statistical maximum a posteriori-GGMRF approach to remove inconsistencies from the labeled image.

The whole-slide image of the histopathology contains enormous amount of information. Therefore, the manual examination of histopathological images is a laborious and time-consuming task for the pathologist. Given that there is a growth in the number of cancer cases, rapid and accurate evaluation is not feasible. Our work addresses this limitation by proposing a computational framework for the automatic analysis of histopathology images with good classification accuracy. Our framework can also be used for other cancer types.

Our approach had a high patchwise accuracy (Tables 2–5). This demonstrates that our approach can identify tissue type from patches with different sizes. For the patchwise accuracy, the best patch size was 35 × 350. We noted that patches with size 350 × 350 had better patchwise accuracy for the cases of fat, parenchyma, and clear cell papillary RCC. In the case of clear cell RCC, the same patchwise accuracy could be obtained for sizes 350 × 350 and 450 × 450, while 350 × 350 size had better accuracy than 250 × 250 size. The results in Table 6 establish the generalizability of our methodology, as we obtained a good classification performance for an external dataset. Therefore, this paper is different from our previous work in terms of the amount of validation and assessing the generalizability of the model.

The ensemble-pyramidal approach allowed processing the pathology image at various spatial scales, and the final classification was obtained by combining the results of the three CNN. Therefore, we obtain better results than with just one CNN. Furthermore, the pixelwise classification gives the pathologist the opportunity to identify pathology regions at small scale. For the pixelwise accuracy, the best patch size was 35 × 350. Patches with size 350 × 350 had better pixelwise accuracy for the cases of parenchyma and clear cell RCC. For fat and clear cell papillary RCC, the same pixelwise accuracy could be obtained for sizes 350 × 350 and 450 × 450, while 350 × 350 size had better accuracy than 250 × 250 size.

For all tissue types, the fusion of the CNN pixel labels resulted in a new labeling for the pixels that had better pixelwise accuracy (Tables 2–5). The highest pixelwise accuracy could be obtained from the fat cases and the lowest from the clear cell papillary RCC and clear cell RCC. We concluded that each patch size taught the CNN to extract new features related to the underlying texture of the tissue and the combination between CNN decisions helped in improving the final classification.

Applying GGMRF smoothing improved the labeling of the tested slide image because it worked on removing inconsistencies (Table 7). Furthermore, our approach showed superior performance in comparison with

Table 2
The average of the patchwise accuracy and pixelwise accuracy for fat cases in the second dataset.

	Patch size			
	250 × 250	350 × 350	450 × 450	Fusion of the CNN results
Average patchwise accuracy	0.87 ± 0.04	0.90 ± 0.11	0.89 ± 0.05	
Average pixelwise accuracy	0.89 ± 0.01	0.91 ± 0.08	0.91 ± 0.11	0.93 ± 0.02

Table 3
The average of the patchwise accuracy and pixelwise accuracy for parenchyma cases in the second dataset.

	Patch size			
	250 × 250	350 × 350	450 × 450	Fusion of the CNN results
Average patchwise accuracy	0.85 ± 0.11	0.88 ± 0.21	0.87 ± 0.03	
Average pixelwise accuracy	0.88 ± 0.05	0.90 ± 0.08	0.89 ± 0.16	0.92 ± 0.03

Table 4
The average of the patchwise accuracy and pixelwise accuracy for clear cell papillary RCC cases in the second dataset.

	Patch size			
	250 × 250	350 × 350	450 × 450	Fusion of the CNN results
Average patchwise accuracy	0.84 ± 0.14	0.87 ± 0.06	0.86 ± 0.07	
Average pixelwise accuracy	0.86 ± 0.11	0.90 ± 0.05	0.90 ± 0.21	0.91 ± 0.01

Table 5
The average of the patchwise accuracy and pixelwise accuracy for clear cell RCC cases in the second dataset.

	Patch size			
	250 × 250	350 × 350	450 × 450	Fusion of the CNN results
Average patchwise accuracy	0.83 ± 0.20	0.86 ± 0.01	0.86 ± 0.02	
Average pixelwise accuracy	0.85 ± 0.05	0.89 ± 0.04	0.88 ± 0.11	0.91 ± 0.03

Table 6
The average of the patchwise accuracy and pixelwise accuracy for the external NIH dataset. This dataset consists of clear cell RCC slide only.

	Patch size			
	250 × 250	350 × 350	450 × 450	Fusion of the CNN results
Average patchwise accuracy	0.82 ± 0.32	0.84 ± 0.20	0.85 ± 0.01	
Average pixelwise accuracy	0.84 ± 0.03	0.87 ± 0.11	0.87 ± 0.21	0.90 ± 0.20

Table 7
Quantitative comparison between our method and other deep learning models using the second dataset that has tissue distribution as follows: 20% fat, 20% parenchyma, 40% clear cell papillary RCC, and 40% clear cell RCC.

	Average pixelwise accuracy
ResNet18	0.89 ± 0.07
ResNet34	0.88 ± 0.12
Proposed without GGMRF smoothing	0.91 ± 0.02
Proposed with GGMRF smoothing	0.92 ± 0.02

popular CNN architectures such as Resnet 18 and Resnet 34, demonstrating that the idea of combining CNN decisions is beneficial for obtaining improved classification.

4. Conclusions

In this work, we evaluated the performance of an automated classification framework in predicting kidney cancer. In this study, we performed more validation than our previous work.[13] Furthermore, validating our model by external data establish the generalizability of our system. Or system can classify between clear cell RCC and clear cell papillary RCC and showed that deep learning techniques can work well at classifying cancer subtypes with morphologic overlap. Given the success of deep learning in computational pathology, pathologists should appreciate the outcomes of using deep learning. Deep learning reduces the workload on pathologists and in contrast to manual examination, which is susceptible to subjectivity, deep learning produces reproducible results. Furthermore, our work produced precise dense classification in the spatial domain of the histopathology image by introducing the notion of pixelwise classification. Pixelwise classification gives the pathologist the opportunity to identify regions on a small scale.

Financial support and sponsorship

Nil.

Author contributions

	Contributor 1	Contributor 2	Contributor 3	Contributor 4	Contributor 5	Contributor 6
Concepts	✓	✓	✓		✓	
Design	✓	✓			✓	✓
Definition of intellectual content				✓	✓	✓
Literature search	✓	✓				
Clinical studies				✓		✓
Experimental studies	✓	✓	✓		✓	
Data acquisition				✓		✓
Data analysis	✓	✓	✓			
Statistical analysis	✓	✓	✓			
Manuscript preparation	✓	✓				✓
Manuscript editing	✓	✓			✓	✓
Manuscript review				✓	✓	✓
Guarantor						✓

Declaration of Competing Interest

There are no conflicts of interest.

Acknowledgments

None.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jpi.2022.100093.

References

1. Worldwide cancer data: Global cancer statistics for the most common cancers [Internet]. London: World Cancer Research Fund | American Institute for Cancer Research [cited 2020 Jun 29]. Available from: https://www.wcrf.org/dietandcancer/cancer-trends/worldwide-cancer-data.
2. Key statistics about kidney cancer [Internet]. American Cancer Society [updated 2020 Feb 1; cited 2020 Jun 29]. Available from: https://www.cancer.org/cancer/kidney-cancer/about/key-statistics.html.
3. Znaor A, Lortet-Tieulent J, Laversanne M, Jemal A, Bray F. International variations and trends in renal cell carcinoma incidence and mortality. Eur Urol 2015;67:519–530.
4. Kovacs G, Akhtar M, Beckwith BJ, et al. The Heidelberg classification of renal cell tumours. J Pathol 1997;183:131–133.
5. Zhou H, Zheng S, Truong LD, Ro JY, Ayala AG, Shen SS. Clear cell papillary renal cell carcinoma is the fourth most common histologic type of renal cell carcinoma in 290 consecutive nephrectomies for renal cell carcinoma. Hum Pathol 2014;45:59–64.
6. Komura D, Ishikawa S. Machine learning methods for histopathological image analysis. Comput Struct Biotechnol J 2018;16:34–42.
7. Dimitriou N, Arandjelovic O, Caie PD. Deep learning for whole slide image analysis: an overview. Front Med (Lausanne) 2019;6:264.
8. Serag A, Ion-Margineanu A, Qureshi H, et al. Translational AI and deep learning in diagnostic pathology. Front Med (Lausanne) 2019;6:185.
9. Srinidhi CL, Ciga O, Martel AL. Deep neural network models for computational histopathology: a survey. Med Image Anal 2020;67, 101813.
10. Han Z, Wei B, Zheng Y, Yin Y, Li K, Li S. Breast cancer multi-classification from histopathological images with structured deep learning model. Sci Rep 2017;7:4172.
11. Khosravi P, Kazemi E, Imielinski M, Elemento O, Hajirasouliha I. Deep convolutional neural networks enable discrimination of heterogeneous digital pathology images. EBioMedicine 2018;27:317–328.
12. Tabibu S, Vinod PK, Jawahar CV. Pan-renal cell carcinoma classification and survival prediction from histopathology images using deep learning. Sci Rep 2019;9:10509.
13. Abdeltawab H, Khalifa F, Ghazal M, Cheng L, Gondim D, El-Baz A. A pyramidal deep learning pipeline for kidney whole-slide histology images classification. Sci Rep 2021;11(1):1–9.
14. National Cancer Institute. GDC Data Portal. Retrieved from: https://portal.gdc.cancer.gov/.
15. Zuiderveld K. Contrast limited adaptive histogram equalization. In: *Paul SH, ed.* Graphics Gems IV. San Diego: Academic Press Professional, Inc; 1994. p. 474–485.
16. Bouman C, Sauer K. A generalized Gaussian image model for edge-preserving MAP estimation. IEEE Trans Image Process 1993;2:296–310.
17. Besag J. On the statistical analysis of dirty pictures. J R Stat Soc Ser B Methodol 1986;48:259–279.
18. Abadi M, Barham P, Chen J, et al. TensorFlow: a system for large-scale machine learning. Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation. Savannah, GA, USA: USENIX Association; 2016. p. 265–283.