CROPro: a tool for automated cropping of prostate magnetic resonance images

Alexandros Patsanis,[a,*] Mohammed R. S. Sunoqrot,[a,b]
Tone F. Bathen,[a,b] and Mattijs Elschot[a,b,*]

[a]Norwegian University of Science and Technology, Department of Circulation and Medical Imaging, Faculty of Medicine and Health Sciences, Trondheim, Norway
[b]St. Olavs Hospital, Trondheim University Hospital, Department of Radiology and Nuclear Medicine, Trondheim, Norway

Abstract

Purpose: To bypass manual data preprocessing and optimize deep learning performance, we developed and evaluated CROPro, a tool to standardize automated cropping of prostate magnetic resonance (MR) images.

Approach: CROPro enables automatic cropping of MR images regardless of patient health status, image size, prostate volume, or pixel spacing. CROPro can crop foreground pixels from a region of interest (e.g., prostate) with different image sizes, pixel spacing, and sampling strategies. Performance was evaluated in the context of clinically significant prostate cancer (csPCa) classification. Transfer learning was used to train five convolutional neural network (CNN) and five vision transformer (ViT) models using different combinations of cropped image sizes (64×64, 128×128, and 256×256 pixels^2), pixel spacing (0.2×0.2, 0.3×0.3, 0.4×0.4, and 0.5×0.5 mm^2), and sampling strategies (center, random, and stride cropping) over the prostate. T2-weighted MR images ($N = 1475$) from the online available PI-CAI challenge were used to train ($N = 1033$), validate ($N = 221$), and test ($N = 221$) all models.

Results: Among CNNs, SqueezeNet with stride cropping (image size: 128×128, pixel spacing: 0.2×0.2 mm^2) achieved the best classification performance (0.678 ± 0.006). Among ViTs, ViT-H/14 with random cropping (image size: 64×64 and pixel spacing: 0.5×0.5 mm^2) achieved the best performance (0.756 ± 0.009). Model performance depended on the cropped area, with optimal size generally larger with center cropping ($\sim$40 cm^2) than random/stride cropping ($\sim$10 cm^2).

Conclusion: We found that csPCa classification performance of CNNs and ViTs depends on the cropping settings. We demonstrated that CROPro is well suited to optimize these settings in a standardized manner, which could improve the overall performance of deep learning models.

Keywords: deep learning; image cropping; image processing; prostate cancer; magnetic resonance imaging.

Paper 22038GRR received Feb. 10, 2022; accepted for publication Feb. 9, 2023; published online Mar. 7, 2023.

1 Introduction

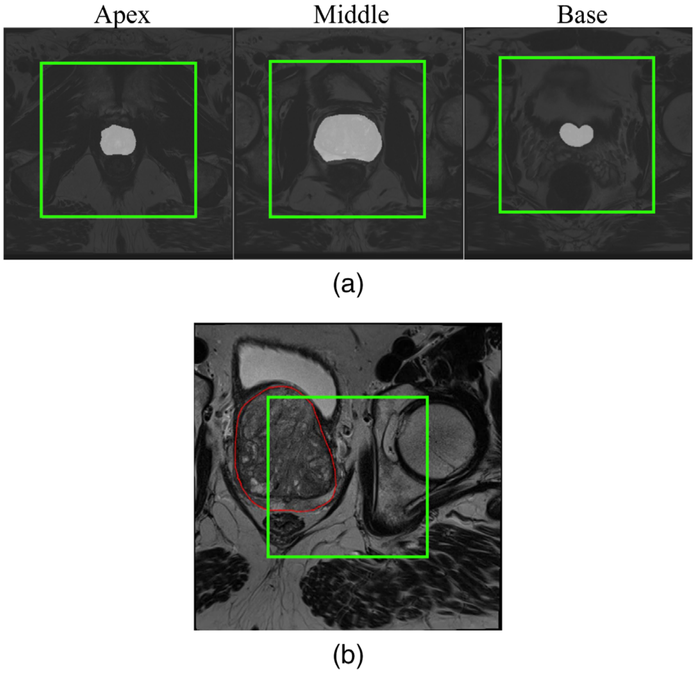
Prostate cancer (PCa) is the fifth cause of death in men and the second most common cancer worldwide.[1] The current diagnostic procedure for PCa is associated with overdiagnosis leading to overtreatment and misdiagnosis of PCa.[2] Magnetic resonance imaging (MRI) is used to assist the biopsy procedure when PCa is suspected.[3] Multiparametric MRI (mpMRI) can improve the detection rate of clinically significant PCa (csPCA) and reduce the overdiagnosis of insignificant

*Address all correspondence to Alexandros Patsanis, alexandros.patsanis@ntnu.no; Mattijs Elschot, mattijs.elschot@ntnu.no

PCa.[2,4] In addition, the combination of mpMRI with computer-aided diagnosis (CAD) systems can contribute to improving decision-making.[5] Recently, deep learning has gained significant attention for performing computer vision tasks, such as segmentation, classification, and object recognition.[6–8] Convolutional neural networks (CNNs) have shown high performance in medical imaging tasks, such as classification of csPCa,[9] breast cancer,[10] lung nodules,[11] and brain tumors.[12] Vision transformers[13] (ViTs) have been shown to outperform conventional CNN models in image classification,[14] semantic segmentation,[15] and 3D object recognition.[16] More recently, ViTs have also shown competitive performance for medical imaging tasks.[17–19] Training deep learning models requires a large, annotated dataset. Recent work[20] has shown that preprocessing of MRI images, such as denoising, MR bias field correction, co-registration, and standardization, improves the performance of classification and segmentation models. However, a subject that has received less attention is how the cropping of these preprocessed images impacts network performance.

Medical images vary in size depending on modality and application, but typically consist of thousands of pixels per slice. Although most deep learning models are adaptable, they have been developed and tested for images of certain sizes, e.g., 32 × 32,[21] 64 × 46,[22] 128 × 128,[23] 256 × 256,[24,25] or images of 512 × 512[26] pixels. This forces users to adjust the input to different models.[27] At the same time, the balance between foreground pixels, representing the region of interest (ROI), and background pixels, representing the region around the ROI, is paramount for deep learning models to achieve robust and accurate results.[28]

For PCa, due to input constraints, it is challenging for most deep learning models to obtain an image that contains the entire prostate with a balanced pixel distribution, as shown in Fig. 1(a). One solution is to crop the image to the region containing the ROI to achieve a better balance between foreground and background pixels. ROIs can be cropped manually or automatically.



(a)

(b)

Fig. 1 (a) Prostate slices from the apex, middle, and base of a patient with an original image size of 384 × 384. The slices have a pixel spacing of 0.5 mm × 0.5 mm, and the cropped square box (256 × 256, in green) is centered on the midpoint of the prostate. The foreground and background pixels are unbalanced, resulting in an image that contains little information about the prostate. (b) A middle slice from a patient with benign prostatic hyperplasia, in which cropping to the center of the image fails. The area surrounded by red represents the segmented prostate gland. Image size is 384 × 384, pixel spacing is 0.5 mm × 0.5 mm, and cropped area is 256 × 256. The image was acquired at St. Olavs Hospital, Trondheim University Hospital, Norway. Use was approved by the institutional review board and The Regional Committee for Medical and Health Research Ethics (REC Central Norway, identifier 2017/576, 2013/1869).

Manual cropping is a tedious and time-consuming task. Therefore, automated cropping methods are more commonly used for tasks that require large numbers of images, such as training deep learning models. The most common approach is center cropping, which assumes that the ROI is located in the center of the image. However, due to differences in image acquisition protocols, there is a risk that the ROI will be cropped inaccurately, and the assumption that the prostate is always located in the center is not always true, as shown in Fig. 1(b).

A robust, flexible, and accurate tool for automated cropping of ROIs, which is compatible with deep learning algorithms and can capture balanced foreground and background pixels in a standardized manner, is currently lacking. Consequently, the contributions of this work are (1) the introduction of CROPro, an open source, publically available tool for standardizing the automated cropping of ROIs regardless of patient health status, ROI size, image size, or pixel spacing and (2) to demonstrate its use by evaluating the impact of image cropping settings in the context of classification of csPCa on T2-weighted (T2W) MR images with CNN and ViT-based models.

2 Related Work

Recently, several deep learning-based approaches have been developed for PCa detection and classification,[9,23,29–36] and the cropping strategy varied between papers. Wang et al.[33] trained a deep CNN to discriminate PCa patients from benign prostate conditions. T2W MR images from 172 patients were used. Each image was downsized to 360 × 360 pixels and then cropped into multiple subimages of 288 × 288 pixels, resulting in a area under the receiver operating characteristic curve (AUC) of 0.84. Yoo et al.[9] proposed a two-level (slice and patient level) automated deep CNN-based pipeline to detect csPCa. Diffusion-weighted MR images (DWI) from 427 patients were used as the dataset. Each DWI slice was resized to a fixed size of 144 × 144 and then center cropped to 66 × 66 pixels, resulting in an AUC of 0.87 and 0.84 at slice and patient level, respectively. Vente at al.[29] used 2D U-Net to both detect and identify the Gleason grade group to estimate lesion aggressiveness on the PROSTATEx-2 challenge dataset. This resulted in a lesion-based weighted kappa of 0.13 ± 0.27. Cropped ROIs of 90 mm × 90 mm × 80 mm, with visual verification of the prostate gland presence within the cropped images were resized to 192 × 192 × 32 voxels for training. Yang et al.[34] developed a deep CNN for PCa detection using mpMRI from 780 patients. They cropped the prostate gland using the prostate segmentation mask with a fixed bounding box of 196 × 196 × 16 voxels and spacing of 0.4 mm × 0.4 mm × 0.5 mm and achieved an AUC of 0.96. Saha et al.[35] proposed two parallel 3D CNNs for patient-level detection of csPCa with an AUC of 0.882 ± 0.030 trained on 1584 MRI scans and tested on 486 scans with PI-RADS v2 annotation, with an input ROI of 144 × 144 × 18 voxels for the detection model and 112 × 112 × 12 voxels for a residual classifier. In their study, the residual classifier takes multichannel batches of 64 × 64 × 8 voxels with a stride of 16 (in-plane) and 4 voxels (through-plane) as input to generate a malignant score per image patch, which is fused with the detection model to identify csPCa. 296 scans with external biopsy confirmation were used to test agreement between model, radiologists ($\text{kappa} = 0.51 \pm 0.4$) and pathologists ($\text{kappa} = 0.56 \pm 0.6$). The input was resampled to a common resolution of 0.5 mm × 0.5 mm × 3.6 mm. Hosseinzadeh at al.[36] used U-Net for detection and localization of csPCa (PI-RADS ≥ 4) with an AUC of 0.88, where input images were resampled to 0.5 mm × 0.5 mm and then cropped by 9.6 × 9.6 cm around the center. A dataset of 2734 mpMRI scans was used for training and testing.

Pachetti et al.[23] evaluated ViTs for classification low-grade versus high-grade PCa lesions (Gleason score $\leq 3 + 4$ versus $\geq 4 + 3$) trained from scratch on ProstateX-2 with axial volumetric T2W images, with a mean AUC of 0.775 ± 0.094 on fivefold cross validation. All images were scaled to the largest image size in the dataset (384 × 384) and cropped with a fixed size of 128 × 128, assuming that the prostate was located in the center of the image. Lesion volumes of size 128 × 128 × 5 were used for analysis. In summary, the size of the input images to the models reported above varied and none of the studies reported optimization of the cropping strategy.

An automated tool for standardized cropping of prostate images would presumably help increase model robustness, comparability, and generalizability between centers. In their PCa detection system, Yang et al.[37] proposed to automatically crop the image using a regression

CNN model to prune a square region containing the entire prostate gland. For this purpose, T2W original images were paired with manually labeled square bounding boxes. Although the subsequent step in their detection network can handle varying image sizes resulting from this approach, it is limited to the size of the manual square bounding boxes and does not take into account pixel spacing, which can vary from patient to patient and protocol to protocol. Zaridis et al.[38] proposed an automated prostate gland cropping approach based on a U-Net to predict an amorphous region around the prostate. A bounding box was determined using the minimum and maximum coordinates of the amorphous mask and resampled to 256×256 pixels. This approach improved the deep learning-based segmentation accuracy with up to 8.9%. However, resampling an image to a fixed size will cause the image to be scaled and stretched, which may affect model performance since the original image is modified.

3 Proposed Method

There is a need for an automated cropping tool for deep learning applications that works under the following conditions:

- *Image size*. The tool can handle arbitrary-sized images as input and return cropped images of chosen size as output.
- *Image resolution*. The tool can handle images with any resolution (pixel spacing) as input and return cropped images with chosen resolution as output.
- *Coverage*. The tool offers different sampling strategies to cover the complete ROI.

Meeting these conditions enables flexible image sampling, compatible with any deep learning method and allows optimizing the balance between foreground and background pixels for the problem at hand.

CROPro, our proposed tool, can crop images of any size and resolution using three sampling strategies: center cropping, random cropping, and stride cropping. Each of these strategies is based on a (manually or automatically) segmented ROI, e.g., the whole prostate gland or a suspected lesion. The general method is illustrated in Fig. 2 and outlined in detail in the next sections.

3.1 *Image Resampling*

Medical images are acquired at different institutions with varying scanners and slightly different scanning protocols. Consequently, the image resolution or pixel spacing (measured in mm) of the input images in a dataset may be different. This is not ideal for deep learning purposes as the model performance can be affected if a trained model is tested on data with a different resolution. Resampling the images to the same pixel spacing helps solving this problem. In addition, adjusting the pixel spacing in combination with the crop size can help optimize the balance between foreground and background pixels. Figures 3–6 show different pixel spacings and crop sizes based on the segmentation mask. For example, images of size 128×128 pixels with pixel spacing of $0.5\ \text{mm} \times 0.5\ \text{mm}$ (Fig. 3) have more background pixels than images of size 128×128 pixels with a pixel spacing of $0.4\ \text{mm} \times 0.4\ \text{mm}$ (Fig. 4).

CROPro allows resampling of images with different pixel spacings. The mask image (segmentation) is interpolated using nearest neighbor interpolation, and B-spline interpolation is used to convert the original images into the new pixel space. Both techniques are commonly used for resampling of medical images.[20] The resampling is performed using SimpleITK (Python version 1.2.0).[39]

3.2 *Image Cropping*

CROPro can crop input images to output images of any chosen size with three different sampling techniques (center, random, and stride). In center cropping, a single output image of chosen size is sampled from the center of the segmented ROI. In random cropping, one or more output

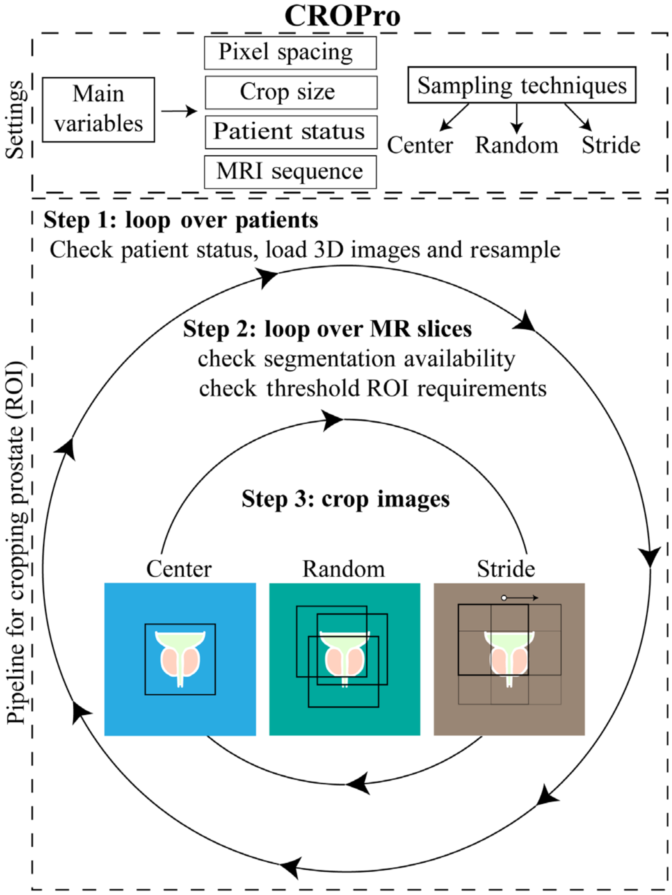

CROPro
Settings
Main variables
Pixel spacing
Crop size
Patient status
MRI sequence
Sampling techniques
Center
Random
Stride
Pipeline for cropping prostate (ROI)
Step 1: loop over patients
Check patient status, load 3D images and resample
Step 2: loop over MR slices
check segmentation availability
check threshold ROI requirements
Step 3: crop images
Center
Random
Stride

Fig. 2 The pipeline of the CROPro tool. Among the different settings, the pixel spacing, the size of the cropped image, and three different sampling techniques can be freely selected, along with several other settings, such as cropping factors depending on the selected sampling technique, the patient's health status, the type of MRI sequence, and the type of image to be stored. Depending on the patient's health status regarding prostate cancer (positive, negative, or unknown), a loop is started for all MRI slices, checking the availability of a segmentation mask and the fulfillment of threshold ROI size criteria (i.e., cropped area > minimum ROI area) before applying the cropping techniques.

images of chosen size are sampled from the ROI by setting the center of the cropping mask to a random pixel in the ROI. In CROPro, the number of randomly cropped samples (N_{samples}) is controlled by the parameter C_{random} in the following equation:

$$N_{\text{samples}} = \frac{N_{\text{roi}}}{N_{\text{crop}}} * C_{\text{random}}, \tag{1}$$

where N_{roi} is the number of pixels in the ROI and N_{crop} is the number of pixels in the cropped output images, e.g., 64×64, 128×128, or 256×256. C_{random} is an empirically chosen factor that controls the number of samples required to cover the ROI and is only used in random cropping mode. Setting C_{random} too low could result in undersampling of the ROI, whereas too high values of C_{random} could result in oversampling. For example, with $N_{\text{roi}} = 10{,}000$ and $N_{\text{crop}} = 128 \times 128$, the division in Eq. (1) is equal to 0.610. Setting C_{random} to 10 will then result in N_{samples} equal to 6.10. N_{samples} is rounded down to the nearest integer, resulting in six samples in this example. Of note, the choice of C_{random} is not affected by pixel spacing, ROI size, and crop size, as N_{samples} scales automatically with these parameters.

In stride cropping, one or more output images of chosen size are sampled from a rectangular box around the ROI by systematically moving from top left to bottom right, skipping C_{stride}

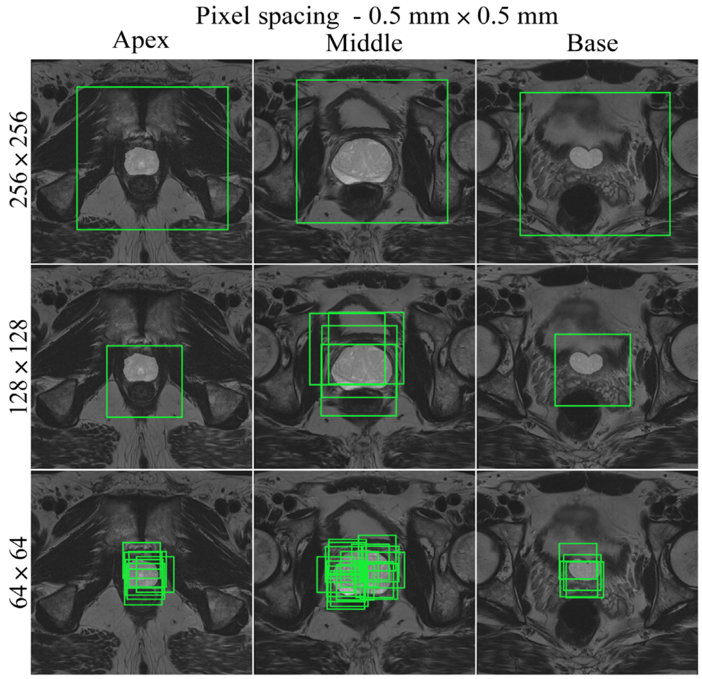


Fig. 3 Slices through the apex, middle, and base of the prostate for a negative patient, using the random cropping function. Three different cropped image sizes (64 × 64, 128 × 128, and 256 × 256) are shown, overlaid with cropping boxes in light green color. The images have 0.5 mm × 0.5 mm pixel spacing.

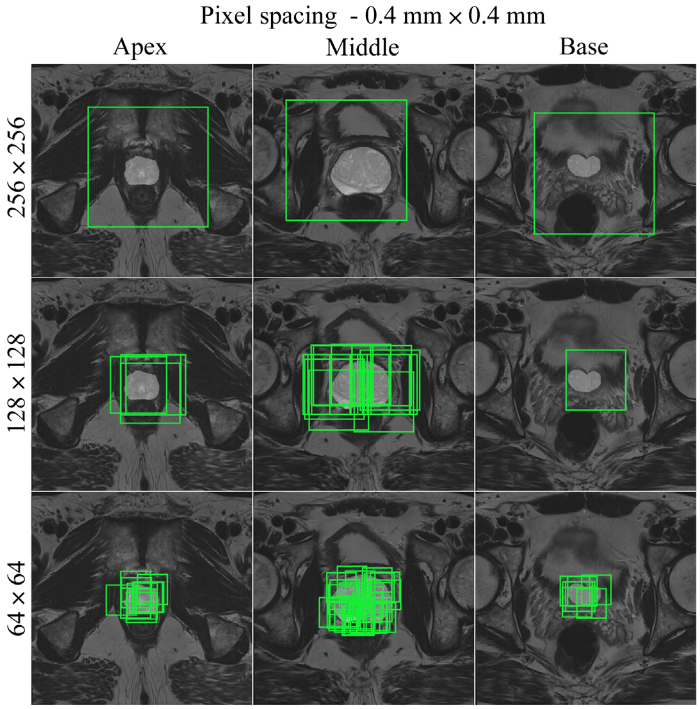


Fig. 4 Slices through the apex, middle, and base of the prostate for a negative patient, using the random cropping function. Three different cropped image sizes (64 × 64, 128 × 128, and 256 × 256) are shown, overlaid with cropping boxes in light green color. The images have 0.4 mm × 0.4 mm pixel spacing.

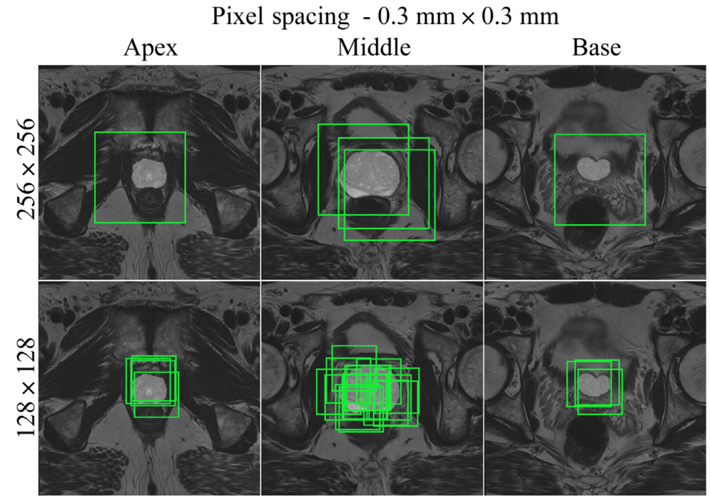


Fig. 5 Slices through the apex, middle, and base of the prostate for a negative patient, using the random cropping function. Two different cropped image sizes (128 × 128 and 256 × 256) are shown, overlaid with cropping boxes in light green color. The images have 0.3 mm × 0.3 mm pixel spacing.

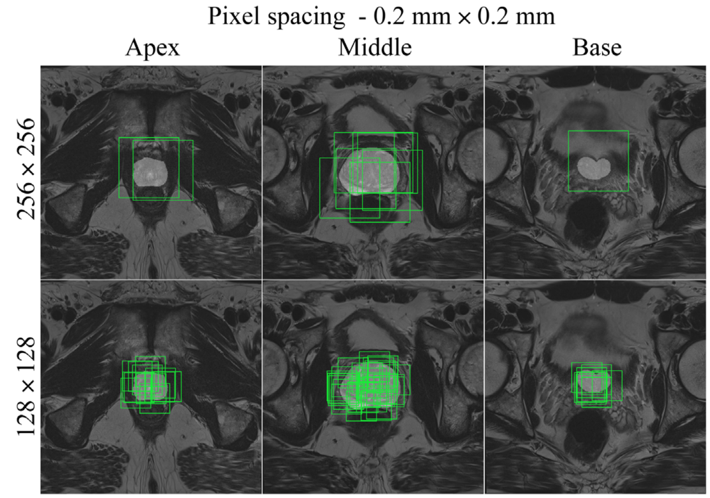


Fig. 6 Slices through the apex, middle, and base of the prostate for a negative patient, using the random cropping function. Two different cropped image sizes (128 × 128 and 256 × 256) are shown, overlaid with cropping boxes in light green color. The images have 0.2 mm × 0.2 mm pixel spacing.

pixels in each direction. Consequently, the number of samples is given by the following equation:

$$N_{\text{samples}} = \frac{H_{\text{box}} - H_{\text{im}}}{N_{\text{stride}+1}} * \frac{W_{\text{box}} - W_{\text{im}}}{N_{\text{stride}+1}}, \tag{2}$$

where H_{box} and W_{box} are the height and width, respectively, of a box covering the entire ROI rounded up to the nearest multiple of C_{stride}, and H_{im} and W_{im} are the chosen height and width of the output image.

3.3 *Image Selection*

To present the deep learning network with only the information relevant to the task at hand, we often want to select the images that contain a significant part of the ROI. Images that have limited task-related information, e.g., containing only a very small part of a lesion, could potentially harm the training procedure. To ensure that images with too little task-related information are not included in the training set, we introduce a threshold parameter $C_{\mathrm{min\_area}}$, which specifies the minimum area (in mm^2) of ROI that needs to be present in a cropped image. The value of $C_{\mathrm{min\_area}}$ depends on the clinical problem and needs to be determined and adjusted per use case. In CROPro, a cropped image will only be processed if Eq. (3) is true:

$$C_{\mathrm{min\_area}} < N_{\mathrm{roi}} * \mathrm{pixel\_spacing}^2. \tag{3}$$

4 Experiments and Results

The Ubuntu 18.04.5 LTS operating system with a single NVIDIA Tesla V100S PCIe 32 GB GPU was used for all experiments. Using transfer learning, five CNN-based models and five ViT-based models[13] were trained, validated, and tested for image-level csPCa classification. Our hypothesis is that the performance of the models depends on the CROPro settings for cropping of the input images. The code is written in Python (version 3.6.9).

4.1 *Dataset*

The publicly available PI-CAI challenge dataset[40] ($N = 1500$) was used to train, validate, and test the 10 deep learning models. 25 cases were excluded, due to having more than one scan per patient ($N = 24$) or lacking clinical information ($N = 1$). The cases were classified as positive ($N = 425$) or negative ($N = 1050$) for csPCa based on International Society of Urological Pathology criteria for grading of PCa.[41] The dataset was divided into a training set (70%, $N = 1033$, 736 negative and 297 positive cases), validation set (15%, $N = 221$, 157 negative and 64 positive cases), and a test set (15%, N = 221, 157 negative and 64 positive cases). The split was done randomly for both negative and positive cases, with the exception that the validation and test sets were assigned only positive cases with human labels. In contrast, the training set contained a mix of human ($N = 92$) and AI-labeled ($N = 205$) positive cases. The T2W MR images were normalized using AutoRef [42] and used as input for the networks. The pixel spacing of the original images varied from 0.23 mm × 0.23 mm to 0.78 mm × 0.78 mm, and interslice spacing ranged from 2.2 to 5.0 mm. Image size varied from 256 × 256 to 1024 × 1024. In this study, PI-CAI data version 1.0 was used for all experiments.

4.2 *CROPro Parameters*

The following CROPro settings were investigated for cropping the T2W MR images in the training set: pixel spacing 0.2 mm × 0.2 mm, 0.3 mm × 0.3 mm, 0.4 mm × 0.4 mm, and 0.5 mm × 0.5 mm; image size 64 × 64, 128 × 128, and 256 × 256 pixels; and sampling with center cropping, random cropping, and stride cropping. C_{random} was empirically set to 12 and C_{stride} to 32. $C_{\mathrm{min\_area}}$ was set to 1 cm^2 (prostate mask) for negative cases, and to 0.2 cm^2 (lesion mask) for positive cases. This means that only cropped images containing more than 1 cm^2 of the prostate ROI were used for the negative cases, and cropped images with more than 0.2 cm^2 of lesion ROI were used for the positive cases. Segmentations of the first (apex) and last (base) slices of the prostate gland were excluded to avoid bias in AI-based segmentation. Furthermore, segmented lesions were retained only if they overlapped with ≥50% of the prostate mask. During validation and testing, images were systematically sampled using stride cropping (with the same C_{stride} and $C_{\mathrm{min\_area}}$) to ensure that the entire prostate ROI was covered. Table 1 provides information on the datasets generated with the investigated CROPro settings. The combinations of image size 64 × 64 and pixel spacing of 0.3 mm × 0.3 mm and 0.2 mm × 0.2 mm were considered to provide too small images and were excluded from further analysis. Figures 3–6 show three slices (apex, middle, and base) from a selected negative patient using the random cropping technique with different image sizes and different values for pixel spacings. Figure 7 shows an example of a positive patient for a pixel spacing of 0.5 mm × 0.5 mm and different cropped

Table 1 Details about all datasets generated using center, random, and stride cropping with three different cropped sizes (64 × 64, 128 × 128, and 256 × 256) and four different pixels spacings (0.5 mm × 0.5 mm, 0.4 mm × 0.4 mm, 0.3 mm × 0.3 mm, and 0.2 mm × 0.2 mm).

Cropped image (pixels^2)	Pixel space (mm^2)	Train center neg. $(N = 736)$/pos. $(N = 297)$	Train random neg. $(N = 736)$/pos. $(N = 297)$	Train stride neg. $(N = 736)$/pos. $(N = 297)$	Val. stride neg. $(N = 157)$/pos. $(N = 64)$	Test stride neg. $(N = 157)$/pos. $(N = 64)$
256 × 256	0.5 × 0.5	9143/1069	9322/1058	9145/1069	1891/265	1940/288
256 × 256	0.4 × 0.4	9183/1088	12,327/1220	9183/1089	1917/275	1945/319
256 × 256	0.3 × 0.3	9156/1091	22,731/2221	9218/1091	1920/274	1965/320
256 × 256	0.2 × 0.2	5796/843	55,945/5616	18,711/1407	3970/388	4563/537
128 × 128	0.5 × 0.5	8595/1086	34,290/3408	9873/1102	2066/279	2132/331
128 × 128	0.4 × 0.4	5712/826	55,946/5613	15,267/1324	3254/343	3456/458
128 × 128	0.3 × 0.3	1917/212	102,937/10,429	41,719/2667	9059/725	9513/943
128 × 128	0.2 × 0.2	276/16	237,376/24,131	167,816/7489	35,902/2008	37,575/2781
64 × 64	0.5 × 0.5	816/70	150,304/15,255	44,407/2454	9483/664	9783/838
64 × 64	0.4 × 0.4	256/15	237,392/24,118	75,527/3412	16,089/900	16,566/1187

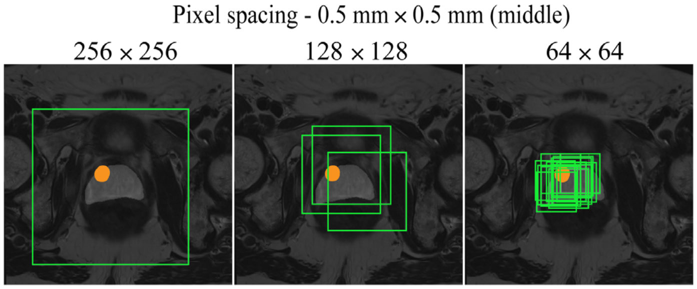


Fig. 7 A slice through the middle of the prostate for a positive patient using the random cropping technique. Three different cropped image sizes (64 × 64, 128 × 128, and 256 × 256) are shown, overlaid with the cropping boxes in light green color. The orange area represents the lesion delineation. The images have 0.5 mm × 0.5 mm pixel spacing.

image sizes. The number of sampled images increases for smaller cropped image sizes, smaller pixel spacings, and larger prostate ROIs.

4.3 *csPCa Classification with CNNs and ViTs*

AlexNet,[43] VGG-19,[44] ResNet50,[8] SqueezeNet,[45] and DesnseNet121[46] and five ViT[13] models were trained, validated, and tested for image-level classification of csPCa. Each ViT model represents a different combination of model size (base, large, and huge)[13] and patch size (14 × 14, 16 × 16, and 32 × 32). Table 2 provides an overview of all deep learning models used in this study. We investigated the impact of the different CROPro settings on model performance.

Overall, 30 different training datasets were investigated for each model, leading to 300 combinations. Each model was trained 5 times for each combination. The cases in the training, validation, and test sets were kept the same to allow for fair comparison between experiments. Each generated dataset was balanced by randomly selecting the same number of negative images as the available number of positive images.

Table 2 An overview of the CNN and ViT-based models used for classification of csPCa. For the ViT models, each model represent a variant of the initial ViT model[13] in terms of model size (parameters) and input patch size (14 × 14, 16 × 16, and 32 × 32). The CNN models were loaded with code from github.com/pytorch. All CNN models can be found here: pytorch.org/vision. The ViT models were loaded with code from github.com/huggingface/transformers. All available ViT models can be found here: huggingface.co/models.

CNN models	ViT models
AlexNet[43]	ViT-H/14 (https://huggingface.co/google/vit-huge-patch14-224-in21k)
VGG-19[44]	ViT-L/32 (https://huggingface.co/google/vit-large-patch32-224-in21k)
ResNet50[8]	ViT-L/16 (https://huggingface.co/google/vit-large-patch16-224-in21k)
SqueezeNet[45]	ViT-B/32 (https://huggingface.co/google/vit-base-patch32-224-in21k)
DesnseNet121[46]	ViT-B/16 (https://huggingface.co/google/vit-base-patch16-224-in21k)

The deep learning models were pretrained with ImageNet[47] on ImageNet-1k (CNNs) and ImageNet-21k (ViTs). Transfer learning of the CNNs was implemented using feature extraction, where only the weights of the last layer related to the prediction were updated. Hyperparameters were the same for all models and experiments: batch size 64; the number of epochs 100 with early stopping and a patience count of 10; Adam optimizer with learning rate 0.001, beta1 0.9 and beta2 0.999. The ViT models were implemented using huggingface transformers library.[48] Hyperparameters were the same for all ViT models and experiments using the class trainer API provided by huggingface: batch size 30, epochs 5, learning rate 0.0002, evaluation strategy with steps, fp16 bit precision, save steps 100, evaluation steps 100, and logging steps 10. PyTorch (version 1.9.1) python library[49] was used for implementation.

4.4 *Statistical Analysis*

Classification accuracy was used as a metric of performance. The mean and standard deviation (SD) of 5 runs for each model on the validation and test sets were reported. The performance of each model was compared to the reference model with image size 256 × 256 and 0.5 mm × 0.5 mm pixel spacing. Statistical differences were assessed using two-sample paired *t*-tests. P-values < 0.05 were considered statistically significant.

4.5 *Results*

In the following section, the results from the test set are presented separately for each of the sampling techniques. The results from the validation set are provided to the reader as Supplementary Material.

4.5.1 *Center cropping*

Table 3 shows the mean ± SD for all CNN-based models and Table 4 for all ViT-based models. Both tables represent different settings with center cropping. For CNN-based models, the highest performance (0.621 ± 0.022) was obtained by SqueezeNet with a cropped image size of 256 × 256 and a pixel spacing of 0.2 mm × 0.2 mm. For each network, the model with the best performance was compared with the reference model. Significant improvements were found for AlexNet and SqueezeNet, but not for VGG-19, ResNet50, and DenseNet121 when using either a smaller cropped image size or a smaller pixel spacing (Fig. 8). For ViT-based models, the highest performance (0.662 ± 0.028) was obtained by ViT-H/14 with a cropped image size of 128 × 128 and a pixel spacing of 0.5 mm × 0.5 mm. The best models performed significantly better than the reference model for ViT-H/14, ViT-L/32 but not for ViT-L/16, ViT-B/32, and ViT-B/16.

Table 3 The test accuracy of the five convolutional neural network models for different CROPro settings with center cropping. The best performing settings are highlighted in bold and significant differences between these models and the reference model are indicated with a "*".

Cropped image (pixels^2)	Pixel space (mm^2)	AlexNet	VGG19	ResNet50	SqueezeNet	DenseNet121
256 × 256	0.5 × 0.5	0.584 ± 0.017	0.600 ± 0.017	0.599 ± 0.023	0.605 ± 0.010	**0.590 ± 0.012**
256 × 256	0.4 × 0.4	0.602 ± 0.019	0.586 ± 0.009	0.561 ± 0.015	0.586 ± 0.018	0.569 ± 0.010
256 × 256	0.3 × 0.3	**0.609 ± 0.009***	**0.607 ± 0.021**	0.585 ± 0.019	0.601 ± 0.013	0.578 ± 0.017
256 × 256	0.2 × 0.2	0.570 ± 0.035	0.564 ± 0.022	0.594 ± 0.030	**0.621 ± 0.022***	0.572 ± 0.021
128 × 128	0.5 × 0.5	0.607 ± 0.021	0.600 ± 0.011	**0.606 ± 0.008**	0.609 ± 0.017	0.582 ± 0.033
128 × 128	0.4 × 0.4	0.552 ± 0.007	0.580 ± 0.021	0.576 ± 0.027	0.599 ± 0.016	0.569 ± 0.012
128 × 128	0.3 × 0.3	0.468 ± 0.008	0.497 ± 0.013	0.510 ± 0.019	0.422 ± 0.016	0.493 ± 0.041
128 × 128	0.2 × 0.2	0.438 ± 0.029	0.436 ± 0.039	0.499 ± 0.014	0.478 ± 0.038	0.517 ± 0.018
64 × 64	0.5 × 0.5	0.488 ± 0.028	0.503 ± 0.015	0.496 ± 0.023	0.484 ± 0.018	0.506 ± 0.020
64 × 64	0.4 × 0.4	0.496 ± 0.004	0.459 ± 0.025	0.508 ± 0.019	0.497 ± 0.032	0.492 ± 0.016

Table 4 The test accuracy of the five vision transformer models for different CROPro settings with center cropping. The best performing settings are highlighted in bold and significant differences between these models and the reference model are indicated with a "*".

Cropped image (pixels^2)	Pixel space (mm^2)	ViT-H/14	ViT-L/32	ViT-L/16	ViT-B/32	ViT-B/16
256 × 256	0.5 × 0.5	0.578 ± 0.036	0.600 ± 0.016	0.581 ± 0.008	0.588 ± 0.040	0.592 ± 0.020
256 × 256	0.4 × 0.4	0.633 ± 0.015	0.624 ± 0.021	0.588 ± 0.015	**0.627 ± 0.013**	0.595 ± 0.028
256 × 256	0.3 × 0.3	0.633 ± 0.036	0.597 ± 0.007	0.578 ± 0.021	0.609 ± 0.029	0.592 ± 0.013
256 × 256	0.2 × 0.2	0.628 ± 0.029	0.557 ± 0.035	0.502 ± 0.005	0.573 ± 0.028	0.566 ± 0.055
128 × 128	0.5 × 0.5	**0.662 ± 0.028***	**0.627 ± 0.010***	**0.594 ± 0.055**	0.605 ± 0.038	**0.604 ± 0.021**
128 × 128	0.4 × 0.4	0.627 ± 0.010	0.577 ± 0.020	0.531 ± 0.036	0.560 ± 0.050	0.559 ± 0.040
128 × 128	0.3 × 0.3	0.566 ± 0.013	0.533 ± 0.027	0.497 ± 0.017	0.529 ± 0.053	0.527 ± 0.015
128 × 128	0.2 × 0.2	0.481 ± 0.018	0.459 ± 0.032	0.427 ± 0.060	0.455 ± 0.043	0.465 ± 0.026
64 × 64	0.5 × 0.5	0.518 ± 0.007	0.517 ± 0.013	0.492 ± 0.024	0.513 ± 0.024	0.504 ± 0.016
64 × 64	0.4 × 0.4	0.462 ± 0.010	0.461 ± 0.027	0.471 ± 0.025	0.471 ± 0.030	0.480 ± 0.022

4.5.2 *Random cropping*

Tables 5 (CNN-based models) and 6 (ViT-based models) show the mean ± SD for all trained models and different settings with random cropping. For CNNs, the highest performance (0.662 ± 0.005) was achieved by ResNet50 with a cropped image size of 256 × 256 and pixel spacing of 0.2 mm × 0.2 mm. The best models performed significantly better than the reference model for all networks except VGG19, when using either a smaller cropped image size or a smaller pixel spacing (Fig. 8). The best models with random cropping performed significantly better than the best models with center cropping for all models except AlexNet and VGG19.

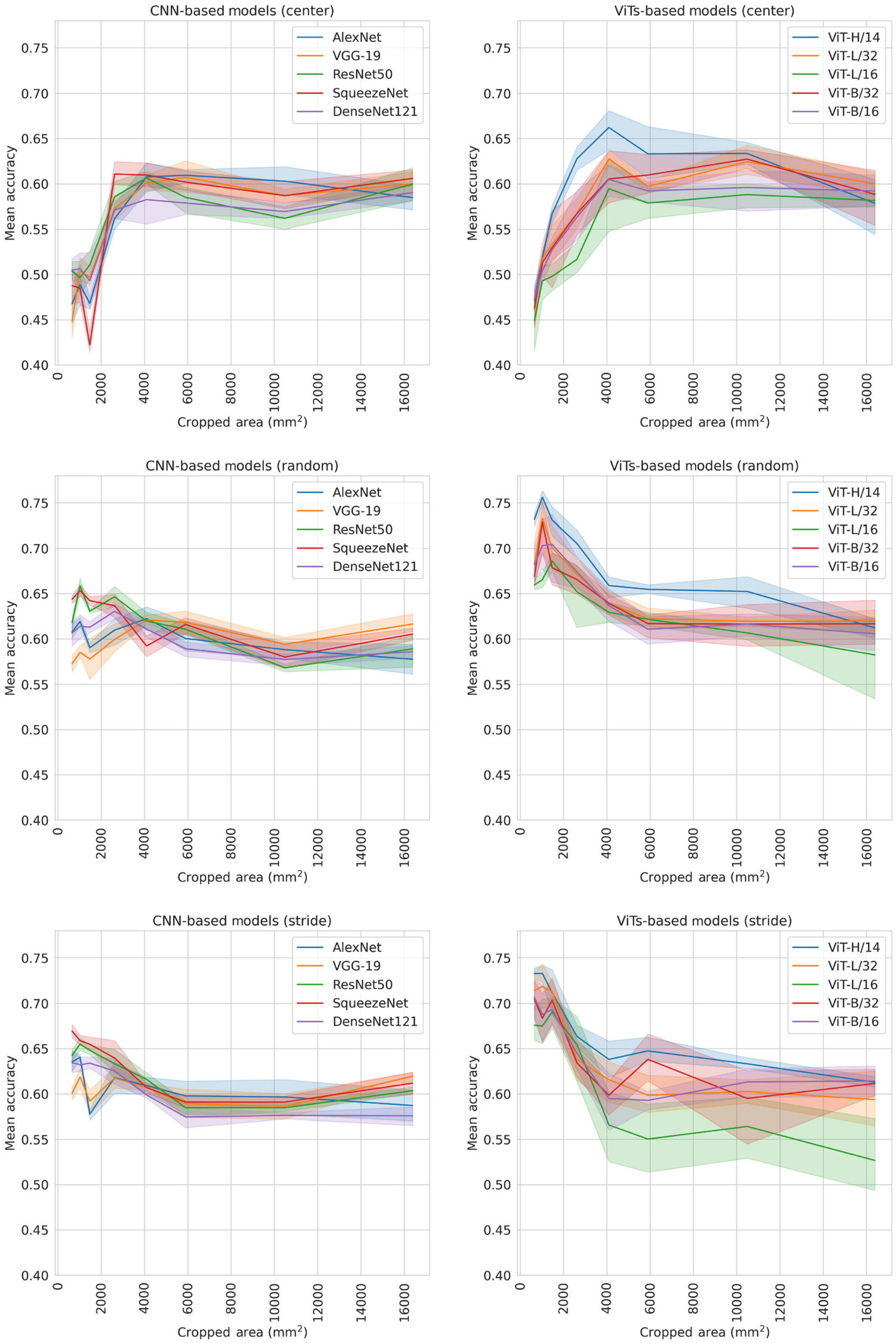


Fig. 8 The performance of each sampling technique (center, random, and stride) for all CNN and ViT-based models, as a function of the area of the cropped images. The solid lines represent the mean accuracy and the shaded areas the 95% confidence intervals.

For ViTs, the highest performance (0.756 ± 0.009) was obtained by ViT-H/14 with a cropped image size of 64×64 and a pixel spacing of $0.5\ \mathrm{mm} \times 0.5\ \mathrm{mm}$. The best models performed significantly better than the reference model for all models. Random cropping performed significantly better than center cropping for all models.

Table 5 The test accuracy of the five convolutional neural network models for different CROPro settings with random cropping. The best performing settings are highlighted in bold and significant differences between these models and the reference model are indicated with a "*".

Cropped image (pixels^2)	Pixel space (mm^2)	AlexNet	VGG19	ResNet 50	SqueezeNet	DenseNet 121
256 × 256	0.5 × 0.5	0.577 ± 0.021	0.616 ± 0.013	0.588 ± 0.023	0.605 ± 0.008	0.585 ± 0.008
256 × 256	0.4 × 0.4	0.588 ± 0.011	0.594 ± 0.012	0.568 ± 0.005	0.579 ± 0.017	0.577 ± 0.007
256 × 256	0.3 × 0.3	0.600 ± 0.021	0.618 ± 0.016	0.610 ± 0.007	0.615 ± 0.011	0.589 ± 0.010
256 × 256	0.2 × 0.2	0.611 ± 0.034	0.607 ± 0.034	**0.662 ± 0.005***	0.650 ± 0.019	**0.630 ± 0.018***
128 × 128	0.5 × 0.5	**0.622 ± 0.014***	**0.620 ± 0.006**	0.619 ± 0.013	0.592 ± 0.015	0.610 ± 0.011
128 × 128	0.4 × 0.4	0.607 ± 0.013	0.591 ± 0.013	0.629 ± 0.011	0.622 ± 0.011	0.630 ± 0.012
128 × 128	0.3 × 0.3	0.590 ± 0.007	0.577 ± 0.028	0.630 ± 0.003	0.642 ± 0.005	0.612 ± 0.008
128 × 128	0.2 × 0.2	0.601 ± 0.018	0.568 ± 0.021	0.622 ± 0.013	0.650 ± 0.003	0.625 ± 0.016
64 × 64	0.5 × 0.5	0.619 ± 0.006	0.585 ± 0.008	0.658 ± 0.010	**0.653 ± 0.008***	0.613 ± 0.017
64 × 64	0.4 × 0.4	0.605 ± 0.007	0.576 ± 0.007	0.613 ± 0.010	0.637 ± 0.003	0.587 ± 0.012

Table 6 The test accuracy of the five vision transformer models for different CROPro settings with random cropping. The best performing settings are highlighted in bold and significant differences between these models and the reference model are indicated with a "*".

Cropped image (pixels^2)	Pixel space (mm^2)	ViT-H/14	ViT-L/32	ViT-L/16	ViT-B/32	ViT-B/16
256 × 256	0.5 × 0.5	0.612 ± 0.014	0.619 ± 0.016	0.582 ± 0.056	0.616 ± 0.030	0.605 ± 0.023
256 × 256	0.4 × 0.4	0.652 ± 0.021	0.619 ± 0.006	0.606 ± 0.010	0.616 ± 0.029	0.616 ± 0.015
256 × 256	0.3 × 0.3	0.654 ± 0.006	0.622 ± 0.014	0.621 ± 0.009	0.616 ± 0.017	0.610 ± 0.019
256 × 256	0.2 × 0.2	0.710 ± 0.033	0.662 ± 0.019	0.653 ± 0.086	0.675 ± 0.016	0.669 ± 0.029
128 × 128	0.5 × 0.5	0.658 ± 0.015	0.638 ± 0.016	0.629 ± 0.014	0.639 ± 0.013	0.636 ± 0.023
128 × 128	0.4 × 0.4	0.700 ± 0.019	0.680 ± 0.028	0.650 ± 0.008	0.655 ± 0.031	0.678 ± 0.030
128 × 128	0.3 × 0.3	0.731 ± 0.020	0.700 ± 0.030	**0.686 ± 0.013***	0.678 ± 0.023	**0.704 ± 0.049***
128 × 128	0.2 × 0.2	0.734 ± 0.007	0.703 ± 0.045	0.663 ± 0.012	0.673 ± 0.015	0.693 ± 0.029
64 × 64	0.5 × 0.5	**0.756 ± 0.009***	**0.732 ± 0.042***	0.665 ± 0.011	**0.728 ± 0.038***	0.703 ± 0.037
64 × 64	0.4 × 0.4	0.729 ± 0.015	0.662 ± 0.021	0.655 ± 0.005	0.664 ± 0.016	0.670 ± 0.038

4.5.3 *Stride cropping*

Tables 7 (CNN-based models) and 8 (ViT-based models) show the mean ± SD for all trained models and different settings with stride cropping. For CNNs, the highest performance (0.678 ± 0.006) was achieved by SqueezeNet with a cropped image size of 128×128 and a pixel spacing of $0.2\text{ mm} \times 0.2\text{ mm}$. The best found models performed significantly better than the reference model when using either a smaller cropped image size or a smaller pixel spacing (Fig. 8) for all models except for VGG19. The best model with stride cropping performed significantly better than the best model with random cropping for AlexNet, DenseNet121, and

Table 7 The test accuracy of the five convolutional neural network models for different CROPro settings with stride cropping. The best performing settings are highlighted in bold and significant differences between these models and the reference model are indicated with a "*".

Cropped image (pixels^2)	Pixel space (mm^2)	AlexNet	VGG19	ResNet 50	SqueezeNet	DenseNet121
256 × 256	0.5 × 0.5	0.587 ± 0.021	0.619 ± 0.006	0.603 ± 0.005	0.611 ± 0.017	0.575 ± 0.013
256 × 256	0.4 × 0.4	0.596 ± 0.025	0.586 ± 0.018	0.584 ± 0.005	0.590 ± 0.007	0.576 ± 0.004
256 × 256	0.3 × 0.3	0.597 ± 0.022	0.589 ± 0.020	0.584 ± 0.011	0.590 ± 0.007	0.574 ± 0.014
256 × 256	0.2 × 0.2	0.642 ± 0.021	**0.629 ± 0.022**	**0.653 ± 0.017***	0.662 ± 0.024	0.642 ± 0.019
128 × 128	0.5 × 0.5	0.609 ± 0.012	0.606 ± 0.005	0.616 ± 0.009	0.606 ± 0.007	0.598 ± 0.005
128 × 128	0.4 × 0.4	0.593 ± 0.011	0.605 ± 0.011	0.612 ± 0.015	0.616 ± 0.011	0.607 ± 0.013
128 × 128	0.3 × 0.3	0.577 ± 0.007	0.591 ± 0.017	0.648 ± 0.006	0.654 ± 0.008	0.633 ± 0.008
128 × 128	0.2 × 0.2	**0.647 ± 0.017***	0.605 ± 0.014	0.649 ± 0.012	**0.678 ± 0.006***	**0.650 ± 0.008***
64 × 64	0.5 × 0.5	0.640 ± 0.004	0.618 ± 0.013	0.655 ± 0.008	0.658 ± 0.006	0.632 ± 0.010
64 × 64	0.4 × 0.4	0.623 ± 0.011	0.595 ± 0.007	0.635 ± 0.004	0.659 ± 0.009	0.618 ± 0.008

Table 8 The test accuracy of the five vision transformer models for different CROPro settings with stride cropping. The best performing settings are highlighted in bold and significant differences between these models and the reference model are indicated with a "*".

Cropped image (pixels^2)	Pixel space (mm^2)	ViT-H/14	ViT-L/32	ViT-L/16	ViT-B/32	ViT-B/16
256 × 256	0.5 × 0.5	0.612 ± 0.007	0.593 ± 0.039	0.526 ± 0.051	0.611 ± 0.017	0.614 ± 0.022
256 × 256	0.4 × 0.4	0.633 ± 0.007	0.602 ± 0.018	0.564 ± 0.045	0.594 ± 0.054	0.613 ± 0.020
256 × 256	0.3 × 0.3	0.647 ± 0.016	0.598 ± 0.026	0.550 ± 0.047	0.638 ± 0.033	0.592 ± 0.013
256 × 256	0.2 × 0.2	0.665 ± 0.025	0.658 ± 0.018	0.678 ± 0.038	0.649 ± 0.026	0.656 ± 0.007
128 × 128	0.5 × 0.5	0.638 ± 0.028	0.615 ± 0.028	0.565 ± 0.049	0.598 ± 0.024	0.594 ± 0.041
128 × 128	0.4 × 0.4	0.661 ± 0.018	0.617 ± 0.040	0.630 ± 0.075	0.619 ± 0.012	0.633 ± 0.039
128 × 128	0.3 × 0.3	0.711 ± 0.033	0.712 ± 0.019	**0.690 ± 0.021***	0.703 ± 0.032	0.693 ± 0.018
128 × 128	0.2 × 0.2	**0.737 ± 0.010***	**0.741 ± 0.022***	0.675 ± 0.009	**0.714 ± 0.032***	**0.735 ± 0.018***
64 × 64	0.5 × 0.5	0.733 ± 0.011	0.718 ± 0.031	0.674 ± 0.032	0.683 ± 0.036	0.687 ± 0.039
64 × 64	0.4 × 0.4	0.728 ± 0.008	0.687 ± 0.023	0.676 ± 0.043	0.694 ± 0.042	0.678 ± 0.025

SqueezeNet, but not for VGG19 and ResNet50. Furthermore, all models performed significantly better with stride cropping than with center cropping, except VGG19. For ViTs, the highest performance (0.741 ± 0.022) was obtained by ViT-L/32 with a cropped image size of 128×128 and a pixel spacing of $0.2\ \mathrm{mm} \times 0.2\ \mathrm{mm}$. The best models performed significantly better than the reference model for all models. The best models with stride cropping performed significantly better than the best models with center cropping, but not than random cropping, which performed significantly better for ViT-H/14.

5 Discussion

In this study, we introduced CROPro, an automated tool for standardized cropping of prostate MRIs to tackle challenges associated with manual preprocessing. We found that carefully optimizing the cropping of the prostate gland from the MR images improved csPCa classification with both CNN and ViT-based models. For both model types, the performance of the tested architectures showed similar dependencies on the cropped image size. For center cropping, the optimal performance was found for medium-sized images ($\sim$40 cm^2). This corresponds to a cropped area that would cover the complete prostate in most images, but does not include too much of the surrounding tissue. For random and stride cropping, the optimal performance was generally associated with smaller images ($\sim$10 cm^2). These would be large enough to cover most lesions while also capturing parts of the surrounding healthy tissue. VGG19 and AlexNet were exceptions to this behavior, but generally showed the lowest performance. It is important to stress that our results could have been different if yet other types of deep learning models, such as U-Net[28] or GANs[50] had been used, if the models had been trained from scratch, if a different classification task had been chosen (e.g., any PCa versus benign patients), and/or if different cropping settings had been tested.

All preprocessing steps for cropping images from the segmented prostate and lesion masks were automated and dependencies on ROI size, input image size and pixel spacing were eliminated. We used transfer learning with five pretrained CNN[51] and five pretrained ViT[13] models for all experiments as a faster alternative to training all models from scratch. This is a valid approach since we only aimed to investigate the impact of cropping parameters on model performance. Although we used state-of-the-art approaches to optimize the pretrained networks, the classification task could potentially benefit from further optimization of the hyperparameters and absolute model performance should therefore be evaluated in this context.

In our experiments, the stride count (C_{stride}) was set to 32 (stride cropping) and C_{random} was set to 12 (random cropping) to balance under and oversampling of the ROI. $C_{\mathrm{min\_area}}$ was set to 0.2 cm^2 for both AI- and human-derived lesion segmentations. The investigated crop sizes (256×256, 128×128, and 64×64 pixels^2) and pixel spacings (0.2×0.2, 0.3×0.3, 0.4×0.4, and 0.5×0.5 mm^2) cover a sensible range of values but could be set to any number.

For the CNN-based models, random and stride cropping generally outperformed center cropping, potentially due to the larger number of images available for training and/or a better ratio of foreground to background pixels. The best performance (0.678 ± 0.006) was obtained with SqueezeNet trained on stride-sampled images of size 128×128 with a pixel spacing of 0.2 mm $\times$ 0.2 mm. This was the highest image resolution tested and in accordance with findings in the literature[52] that show a beneficial effect of higher resolution on classification performance.

For the ViT-based models, random and stride cropping generally outperformed center cropping, in line with results from the CNN-based models. The best performance for the ViT-based models was equal to 0.756 ± 0.009 with ViT-H/14 trained on random-cropped images of size 64×64 with a pixel spacing of 0.5 mm $\times$ 0.5 mm. For all cropping techniques, the best ViT-based model performed better than the best CNN-based model, confirming the potential of ViTs for image classification tasks.

In this study, we used both manual expert and AI segmentations of the prostate glands and lesions as input to CROPro. It should be noted that CROPro works with all types of segmentations, including those automatically generated by deep learning models, such as nnU-Net,[6] which could reduce the workload of radiologists. This approach is in line with the proposal of Vente at al.,[28] who mentioned that the selection of ROIs could be based on the segmentation of the prostate rather than capturing a fixed image center portion.

Several deep learning models have achieved high performance in classifying PCa[9,28,34,36,37] but are dependent on input images of a specific size and pixel spacing. CROPro can be used as a simple tool to prepare the dataset for optimization and fair comparison of these models. In this regard, CAD systems, which have the potential to improve PCa detection, localization, staging, and biopsy targeting, can benefit from tools, such as CROPro to overcome challenges in automated analysis, such as selecting the correct ROI and dealing with input data that varies in pixel spacing and image dimensions.

Although cropping of the prostate appears to be a simple task, it can be quite challenging due to differences in image sources, the size of the prostate, and its location. For example, cropped images missing parts of the prostate may lead to misclassification, as 70% of PCa are located in the peripheral zone of the prostate.[53] In addition, an automated tool should be adaptable and interpretable depending on the patient's health status. For example, during training, it should avoid capturing slices from a positive patient that do not contain lesions. This is critical because these slices may be cancer-free (negative), meaning there is no useful information for classification. Unlike current approaches that focus on sectioning the entire prostate area,[9,28,33–35,37,38] CROPro allows for the assignment of specific ROI types based on health status.

One limitation of our work was that we only tested CROPro with transfer-learned CNN-based and ViT-based models. Furthermore, only image-level accuracy for the test set was reported. Optimization of the csPCa classification task should be done with patient-level accuracy, but was considered outside the scope of this study. Another limitation was that we tested only 256 × 256, 128 × 128, and 64 × 64 input size images, whereas larger or smaller variations could have been considered. Moreover, the 3D MRI volumes were processed in such a way that cropped, 2D images were sampled and saved for each slice. Currently, CROPro does not take into account differences in through-plane resolution (slice thickness) between datasets. The implementation of cropping 3D volumes and handling different through-plane resolutions is subject of future work. Finally, our proposed CROPro tool was implemented and evaluated only for MRI and the task of csPCa classification. Evaluating the tool on other image modalities and/or clinical challenges would be interesting in the future.

6 Conclusion

We proposed and evaluated CROPro, a tool for automated cropping of prostate MRI to bypass manual data preprocessing and improve deep learning performance. We showed that the performance of the csPCa classification task depended on cropping parameters, indicating that fine-tuning these is important for reaching the full potential of deep learning applications.

Disclosures

All authors declare that they have no conflicts of interest.

Acknowledgments

This work was was partly funded by the Research Council of Norway (Grant No. 295013), the Norwegian Cancer Society and Prostatakreftforeningen (Grant No. 215951), and the Liaison Committee between the Central Norway Regional Health Authority (RHA) and the Norwegian University of Science and Technology (NTNU) (Grant Nos. 90265300 and 81770928).

Data, Materials, and Code Availability

Code is available at: https://github.com/alexofficial/CROPro.

References

1. H. Sung et al., "Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA: A Cancer J. Clin.* **71**(3), 209–249 (2021).
2. A. Stabile et al., "Multiparametric MRI for prostate cancer diagnosis: current status and future directions," *Nat. Rev. Urol.* **17**, 41–61 (2020).
3. A. R. Padhani et al., "PI-RADS steering committee: the PI-RADS multiparametric MRI and MRI-directed biopsy pathway," *Radiology* **292**(2), 464–474 (2019).

4. M. van der Leest et al., "Head-to-head comparison of transrectal ultrasound-guided prostate biopsy versus multiparametric prostate resonance imaging with subsequent magnetic resonance-guided biopsy in biopsy-naïve men with elevated prostate-specific antigen: a large prospective multicenter clinical study," *Eur. Urol.* **75**(4), 570–578 (2019).
5. T. Penzkofer et al., "ESUR/ESUI position paper: developing artificial intelligence for precision diagnosis of prostate cancer using magnetic resonance imaging," *Eur. Radiol.* **31**(12), 9567–9578 (2021).
6. F. Isensee et al., "nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation," *Nat. Methods* **18**, 203–211 (2021).
7. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM* **60**, 84–90 (2017).
8. K. He et al., "Deep residual learning for image recognition," arXiv:1512.03385 [cs] (2015).
9. S. Yoo et al., "Prostate cancer detection using deep convolutional neural networks," *Sci. Rep.* **9**, 1–10 (2019).
10. Q. Hu, H. M. Whitney, and M. L. Giger, "A deep learning methodology for improved breast cancer diagnosis using multiparametric MRI," *Sci. Rep.* **10**, 10536 (2020).
11. Y. Li et al., "Lung nodule detection with deep learning in 3D thoracic MR images," *IEEE Access* **7**, 37822–37832 (2019).
12. S. Deepak and P. M. Ameer, "Brain tumor classification using deep CNN features via transfer learning," *Comput. Biol. Med.* **111**, 103345 (2019).
13. A. Dosovitskiy et al., "An image is worth 16 × 16 words: transformers for image recognition at scale," in *ICLR* (2021).
14. C.-F. R. Chen, Q. Fan, and R. Panda, "CrossViT: cross-attention multi-scale vision transformer for image classification," in *Int. Conf. Comput. Vis. (ICCV)* (2021).
15. S. Zheng et al., "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit. (CVPR)*, pp. 6881–6890 (2021).
16. I. Misra, R. Girdhar, and A. Joulin, "An end-to-end transformer model for 3D object detection," in *ICCV* (2021).
17. T. Czempiel et al., "OperA: attention-regularized transformers for surgical phase recognition," *Lect. Notes Comput. Sci.* **12904**, 604–614 (2021).
18. Y. Dai, Y. Gao, and F. Liu, "TransMed: transformers advance multi-modal medical image classification," *Diagnostics* **11**(8), 1384 (2021).
19. S. Li et al., "Medical image segmentation using squeeze-and-expansion transformers," in *Proc. Thirtieth Int. Joint Conf. Artif. Intell. (IJCAI-21)*, Z.-H. Zhou, Ed., pp. 807–815, International Joint Conferences on Artificial Intelligence Organization (2021).
20. S. Masoudi et al., "Quick guide on radiology image pre-processing for deep learning applications in prostate cancer research," *J. Med. Imaging* **8**, 010901 (2021).
21. Y. Lecun et al., "Gradient-based learning applied to document recognition," *Proc. IEEE* **86**, 2278–2324 (1998).
22. D. N. A. Ningrum et al., "Deep learning classifier with patient's metadata of dermoscopic images in malignant melanoma detection," *J. Multidiscip. Healthc.* **14**, 877–885 (2021).
23. E. Pachetti, S. Colantonio, and M. A. Pascali, "On the effectiveness of 3D vision transformers for the prediction of prostate cancer aggressiveness," *Lect. Notes Comput. Sci.* **13374**, 317–328 (2022).
24. O. Kott et al., "Development of a deep learning algorithm for the histopathologic diagnosis and Gleason grading of prostate cancer biopsies: a pilot study," *Eur. Urol. Focus* **7**, 347–351 (2021).
25. W. Alakwaa, M. Nassef, and A. Badr, "Lung cancer detection and classification with 3D convolutional neural network (3D-CNN)," *Int. J. Adv. Comput. Sci. Appl.* **8**(8), 10 (2017).
26. H. N. Veena, A. Muruganandham, and T. Senthil Kumaran, "A novel optic disc and optic cup segmentation technique to diagnose glaucoma using deep learning convolutional neural network over retinal fundus images," *J. King Saud Univ.: Comput. Inf. Sci.* **34**, 6187–6198 (2021).
27. A. Esteva et al., "Deep learning-enabled medical computer vision," *NPJ Digit. Med.* **4**, 1–9 (2021).

28. Z. Khan et al., "Evaluation of deep neural networks for semantic segmentation of prostate in T2W MRI," *Sensors* **20**, 3183 (2020).
29. C. D. Vente et al., "Deep learning regression for prostate cancer detection and grading in bi-parametric MRI," *IEEE Trans. Biomed. Eng.* **68**, 374–383 (2021).
30. R. Cuocolo et al., "Machine learning applications in prostate cancer magnetic resonance imaging," *Eur. Radiol. Exp.* **3**, 35 (2019).
31. Y. Liu and X. An, "A classification model for the prostate cancer based on deep learning," in *10th Int. Congr. Image and Signal Process., BioMed. Eng. and Inf. (CISP-BMEI)*, pp. 1–6 (2017).
32. Y. K. Tsehay et al., "Convolutional neural network based deep-learning architecture for prostate cancer detection on multiparametric magnetic resonance images," *Proc. SPIE* **10134**, 1013405 (2017).
33. X. Wang et al., "Searching for prostate cancer by fully automated magnetic resonance imaging classification: deep learning versus non-deep learning," *Sci. Rep.* **7**, 1–8 (2017).
34. H. Yang et al., "Automatic prostate cancer detection on multi-parametric MRI with hierarchical weakly supervised learning," in *IEEE 18th Int. Symp. Biomed. Imaging (ISBI)*, pp. 316–319 (2021).
35. A. Saha, M. Hosseinzadeh, and H. Huisman, "End-to-end prostate cancer detection in bpMRI via 3D CNNs: effects of attention mechanisms, clinical priori and decoupled false positive reduction," *Med. Image Anal.* **73**, 102155 (2021).
36. M. Hosseinzadeh et al., "Deep learning-assisted prostate cancer detection on bi-parametric MRI: minimum training data size requirements and effect of prior knowledge," *Eur. Radiol.* **32**, 2224–2234 (2021).
37. X. Yang et al., "Co-trained convolutional neural networks for automated detection of prostate cancer in multi-parametric MRI," *Med. Image Anal.* **42**, 212–227 (2017).
38. D. G. Zaridis et al., "A new smart-cropping pipeline for prostate segmentation using deep learning networks," arXiv abs/2107.02476 (2021).
39. B. Lowekamp et al., "The design of SimpleITK," *Front. Neuroinf.* **7**, 45 (2013).
40. A. Saha et al., "Artificial intelligence and radiologists at prostate cancer detection in MRI: the PI-CAI challenge (study protocol) (1.0)," *Zenodo* (2022).
41. L. Egevad et al., "International society of urological pathology (ISUP) grading of prostate cancer: an ISUP consensus on contemporary grading," *APMIS* **124**(6), 433–435 (2016).
42. M. R. S. Sunoqrot et al., "Automated reference tissue normalization of T2-weighted MR images of the prostate using object recognition," *Magn. Reson. Mater. Phys. Biol. Med.* **34**(2), 309–321 (2020).
43. A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," arXiv abs/1404.5997 (2014).
44. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd Int. Conf. Learn. Represent. (ICLR 2015)*, Computational and Biological Learning Society, pp. 1–14 (2015).
45. F. N. Iandola et al., "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size," arXiv abs/1602.07360 (2016).
46. G. Huang et al., "Densely connected convolutional networks," in *2017 IEEE Conf. Comput. Vision and Pattern Recognit. (CVPR)*, Honolulu, HI, USA, pp. 2261–2269 (2017).
47. J. Deng et al., "ImageNet: a large-scale hierarchical image database," in *IEEE Conf. Comput. Vis. and Pattern Recognit.*, IEEE, pp. 248–255 (2009).
48. T. Wolf et al., "Transformers: state-of-the-art natural language processing," in *Proc. 2020 Conf. Empirical Methods in Nat. Lang. Process.: Syst. Demonstrations*, Association for Computational Linguistics, pp. 38–45 (2020).
49. A. Paszke et al., "PyTorch: an imperative style, high-performance deep learning library," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, Curran Associates Inc., Red Hook, NY, USA, pp. 8026–8037 (2019).
50. U. Cem Birbiri et al., "Investigating the performance of generative adversarial networks for prostate tissue detection and segmentation," *J. Imaging* **6**, 83 (2020).
51. S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.* **22**, 1345–1359 (2010).

52. C. F. Sabottke and B. M. Spieler, "The effect of image resolution on deep learning in radiography," *Radiol.: Artif. Intell.* **2**, e190015 (2020).
53. D. Bonekamp et al., "Advancements in MR imaging of the prostate: from diagnosis to interventions," *RadioGraphics* **31**, 677–703 (2011).

Alexandros Patsanis received his BS degree in informatics and communications engineering of technological education in 2015, his pegalological education program from the School of Pedagogical and Technological Education in 2017, and his MS degree in computer science from Uppsala University in 2019. His current research interests include automated CAD systems for prostate cancer detection and localization using deep learning on multiparametric MRI.

Mohammed R. S. Sunoqrot received his BSc and MSc degrees in biomedical engineering from the German Jordanian University in 2014 and the University of Dundee in 2016, respectively, and his PhD in medical technology from Norwegian University of Science and Technology (NTNU) in 2021. He is a postdoctoral researcher at NTNU. His current research focuses on developing computer aided diagnosis systems using multiparametric MRI, radiomics, machine learning, and deep learning.

Tone F. Bathen received her MSc degree in chemistry and her PhD in biomedicine from NTNU. She is a professor in MR technology at NTNU. Her current research focus is within prostate cancer, aiming to implement state-of-the-art technological advancements in imaging for personalized diagnostic and treatment.

Mattijs Elschot received his BSc degree in biomedical engineering and his MSc degree in medical engineering from Eindhoven University of Technology and his PhD from Utrecht University, the Netherlands. He is an associate professor at NTNU. He works on the cutting edge of clinical practice, medical physics, and computer science, with a special focus on prostate cancer imaging and the emerging role of artificial intelligence in this field.