SPECTRAL DOMAIN CONVOLUTIONAL NEURAL NETWORK AND ITS APPLICATIONS

by

Bochen Guan

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Electrical and Computer Engineering)

at the

UNIVERSITY OF WISCONSIN–MADISON

2020

Date of final oral examination: 06/05/20

The dissertation is approved by the following members of the Final Oral Committee:

Prof. William Sethares, Professor, ECE
Prof. Varun Jog, ECE
Prof. Yu Hen Hu, ECE
Prof. Richard Kijowski, Radiology

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

The memory consumption of most Convolutional Neural Network (CNN) architectures grows rapidly with increasing depth of the network, which is a major constraint for efficient network training and inference on modern GPUs with limited memory. Several studies show that the feature maps (as generated after the convolutional layers) are the main bottleneck in this memory problem. Often, these feature maps mimic natural photographs in the sense that their energy is concentrated in the spectral domain.

Although embedding CNN architectures in the spectral domain is widely exploited to accelerate the training process, we demonstrate that it is also possible to use the spectral domain to reduce the memory footprint by proposing a Spectral Domain Convolutional Neural Network (SpecNet) that performs both the convolution and the activation operations in the spectral domain. SpecNet exploits a configurable threshold to force small values in the feature maps to zero, allowing the feature maps to be stored sparsely. SpecNet also employs a special activation function that preserves the sparsity of the feature maps while effectively encouraging the convergence of the network.

The SpecNet methodology can be applied to many different network architectures, and we explicitly explore three state-of-the art implementations: AlexNet, VGG, and DenseNet. The performance of SpecNet is evaluated with several different types of experiments. First, we evaluate SpecNet on three competitive object recognition benchmark tasks (CIFAR-10, SVHN and ImageNet) using each of the three implementations (AlexNet, VGG and DenseNet). SpecNet is able to reduce memory consumption by about 63% without significant loss of performance for all tested network architectures. Second, we apply SpecNet to several medical applications including medical diagnosis and prediction. The use of deep learning for these applications has been less thoroughly explored and is particularly challenging as it often requires analyzing complex abnormalities on multiple slices of different image datasets. Therefore, training on large scale medical application may require large GPU memory. Third, object/logo detection in videos and 3D images can also

be challenging due to memory limitations. Unlike object detection in 2D images, each video contains multiple frames, and information across the frames is crucial for analysis. We test SpecNet in several state-of-the-art detection CNNs and also compare SpecNet to our proposed VLD algorithm.

1 INTRODUCTION

Deep convolutional neural networks have made significant progress on various tasks in recent years [LeCun et al. (2015); Huang et al. (2017); He et al. (2016); Li et al. (2019); Liu et al. (2019)]. Current successful deep CNNs such as ResNet [He et al. (2016)] and DenseNet [Huang et al. (2017)] typically include over 100 layers and require large amounts of training data. Training these models becomes computationally and memory intensive, especially when limited resources are available [Cheng et al. (2017)]. Therefore, it is essential to reduce the memory requirements to allow better network training and deployment, such as applying deep CNNs to embedded systems and cell phones.

Several studies [Jain et al. (2018)] show that the intermediate layer outputs (feature maps) are primary contributors to this memory bottleneck. Existing methods such as model compression [Wu et al. (2016); Courbariaux and Bengio (2016); Hanson and Pratt (1989); Denton et al. (2014)] and scheduling [Pleiss et al. (2017); Chen et al. (2016); Zhang et al. (2019)], do not directly address the storage of feature maps. By transforming the convolutions into the spectral domain, we can reduce the memory requirements of feature maps.

In contrast to [Jain et al. (2018)], which proposes an efficient encoded representation of feature maps in the spatial domain, we exploit the property that the energy of feature maps is concentrated in the spectral domain [Jain et al. (2018)]. Values that are less than a configurable threshold are forced to zero, so that the feature maps can be stored sparsely. We call this approach the Spectral Domain Convolutional Neural Network (SpecNet). In this new architecture, convolutional and activation layers are implemented in the spectral domain. The outputs of convolutional layers are equal to the multiplication of non-zero entries of the inputs and spectral domain kernels. The activation function is designed to preserve the sparsity and symmetry properties of the feature maps in the spectral domain, and to also allow effective derivative computation in back propagation. It is notable that the sparsity of kernels in their spectral representations [Rippel et al. (2015); Ayat et al. (2019)] will use more memory. In contrast, our work is focused on designing

a network to optimize the feature maps and kernels, which in SpecNet are stored in spatial domain.

More specifically, this work contributes the following:

- A new CNN architecture (SpecNet) that performs convolution and activation in the spectral domain. Feature maps are thresholded and compressed to allow reducing model memory by only computing and saving non-zero entries. Kernels are not stored in the spectral domain in order to avoid extra memory consumption.
- A spectral domain activation function is applied to both the real and imaginary parts of the input feature maps, preserving the sparsity property and ensuring effective network convergence during training.
- Extensive experiments are conducted to show the effectiveness of SpecNet using different architectures at multiple computer vision tasks. For example, a SpecNet implementation of DenseNet architecture can reach up to 63% reduction of the memory usage on the SHVN dataset without significant loss of accuracy (95.8% testing accuracy compared with 98.2% accuracy of the original implementation).
- We developed several CNNs for medical applications and for logo detection based on the SpecNet idea. These are evaluated by different types of experiments, datasets and computational environments, and provide strong evidence of the efficacy of SpecNet.

1.1 Related Work

Model Compression

Model compression can be achieved in several ways including quantization, pruning and weight decomposition.

With quantization, the values of filter kernels in the convolutional layers and weight matrices in fully-connected layers are quantized into a limited number of levels. This can decrease the computational complexity and reduce memory cost [Wu et al. (2016); Gupta et al. (2015)]. The extreme case of quantization is binarization [Courbariaux and Bengio (2016); Andri et al. (2018)] which uses only ± 1 to represent all values of the weights, resulting in dramatic memory reduction but risking potentially degraded performance.

Pruning and weight decomposition are other approaches to model compression. The key idea in pruning is to remove unimportant connections. Some initial work [Hanson and Pratt (1989)] focused on using weight decay to sparsify the connections in neural networks while more recent work [Wen et al. (2016); Li et al. (2016)] applied structured sparsity regularizers to the weights. Instead of selecting redundant connections, [Changpinyo et al. (2017)] proposed a compression technique that fixed a random connectivity pattern and required the CNN to train around it. Weight decomposition is based on a low-rank decomposition of the weights in the network. The SVD is an efficient method for decomposition, and has proven to be successful in shrinking the size of the model [Denton et al. (2014)]. Other work [Liu et al. (2015); Zhang et al. (2015)] uses PCA for rank selection. The pruning and weight decomposition attempt to reduce the size of the model so that it is more easily deployed in embedded systems or smart phones. Overall, the aforementioned methods are focused on compressing weights to reduce the model size, and further reduce the size of feature maps. In contrast, SpecNet directly reduces the memory consumption by sparsifying the feature maps and storing them efficiently. The two methods may be combined to save memory.

Memory Sharing

Since the 'life-time' of feature maps (the amount of time data from a given layer must be stored) is different in each layer, it is possible to design data reuse methods to reduce memory consumption. [Pleiss et al. (2017)] observes the feature maps in some layers that are responsible for most of the memory consumption are relatively

cheap to compute. By storing the output of the concatenation, batch normalization and ReLU layers in shared memory, DenseNet can achieve more than 4x memory saving, compared to the original implementation that allocates new memory for these layers. A more general algorithm for designing memory sharing patterns can be found in [Chen et al. (2016)]. It can be applied to CNNs and RNNs with sublinear memory cost compared to the original implementations. Recently, an approach called SmartPool [Zhang et al. (2019)] has been proposed to provide an even more fine-grained memory sharing and reduction strategy.

Representation of Feature Maps in the Spatial Domain

The above methods are not focused on compressing feature maps directly. [Jain et al. (2018)] employed two classes of layer-specific encoding schemes to encode and store the feature maps in the time domain, and to decode the data for back propagation. The additional encoding and decoding process increases the computational complexity. In SpecNet, the architecture is designed for sparse storage of feature maps in the spectral domain, which is more computationally efficient.

[Wang et al. (2017b)] proposed a method to extract intrinsic representations of the feature maps while preserving the discriminability of the features. It can achieve a high compression ratio, but the training process involves a pre-trained CNN and solving an optimization problem. SpecNet does not require additional modules in the training process and is easier to implement.

CNN in the Spectral Domain

Some pilot studies have attempted to combine Fast Fourier Transforms (FFTs) and Wavelet transforms with CNNs [Rippel et al. (2015); Pratt et al. (2017); Mathieu et al. (2013); Fujieda et al. (2017)]. However, most of these works aim to make the training process faster by replacing the traditional convolution operation with the FFT and a product of the inputs and kernel in the spectral domain [Pratt et al. (2017); Mathieu et al. (2013)]. Wavelet CNN [Fujieda et al. (2017)] concatenates the feature maps and multi-resolution features captured by the wavelet transform of the input images

to improve the classification accuracy. These methods do not attempt to reduce memory, and several (such as the Wavelet CNN) require more memory in order to achieve optimal performance. In contrast, SpecNet uses the FFT to reduce memory consumption and its computation complexity depends on certain input parameters. This is quite different from most FFT-based CNN implementations.

1.2 Motivation

Although some progress has been made in achieving memory efficient CNNs, fundamental solutions such as optimizations on feature maps have not been extensively investigated. Research on memory efficient CNNs, on the one hand, have demonstrated approaches using compressed kernels, shared memory implementations, and even specific GPUs to optimize the memory usage of the CNN. However, such methods often apply only to specific situations and hence to particular applications or scenarios such as low testing performance or high-demand and complex computation. Thus there is no low-cost and convenient network that can be used to control memory for both feature maps and kernel weights.

This thesis proposes an efficient and general algorithm, which relies on low-demand computation such as the Fast Fourier transform (FFT) and the Inverse Fast Fourier transform (IFFT) to reduce the memory cost of CNN without sacrificing performance significantly. We call this architecture Spectral Domain Convolutional Neural Network (SpecNet) and examine two variants on AlexNet and LeNet which we call SpecAlex and SpecLeNet.

1.3 Organization

This thesis is organized as follows:

Chapter 2 discusses the detailed structure of SpecNet and shows by experiment that computation and memory reductions can be achieved. A few candidate concepts/algorithms that may be helpful for this problem are introduced.

Chapter 3 develops a fully-automated deep learning-based diagnosis system, by using CNNs to isolate the region of interest on MR images and a classification CNN to detect/predict disease within the isolated region. The structure has been applied to Anterior Cruciate Ligament (ACL) Tears detection and hip bone fracture detection. The network is implemented by SpecNet and testing for each case by this system is on the order of five seconds. The proposed system achieved high diagnosis accuracy (96%) similar to experienced clinical radiologists in detecting knee joint pathology. The system with SpecNet may be easily trained, and used on a desktop in a clinical setting.

Chapter 4 develops a fully-automated deep learning network based on SpecNet to predict the progression of radiographic knee osteoarthritis. There are currently no methods that can accurately predict whether patients with knee osteoarthritis will show disease progression or will not show disease progression over time. The proposed memory efficient system achieves the highest diagnostic performance reported in the literature for the prediction of the progression of radiographic knee osteoarthritis using baseline clinical risk factors and deep learning analysis of knee radiographs.

Chapter 5 develops an algorithm called Video Logo Retrieval (VLD), which is an image-to-video retrieval algorithm based on the spatial distribution of local image descriptors that measure the distance between the query image (the logo) and a collection of down-sampled video images. The proposed algorithm addresses limitations of global feature-based models such as convolutional neural networks as it is flexible and does not require training after the setting of initial hyper-parameters. Besides, Several algorithms show acceptable accuracy but require very large memory consumption, making it hard to achieve real time retrieval in low power devices. We applying SpecNet to the features extractor of VLD and significantly reduce its memory.

Chapter 6 discusses future work and shows a few candidate applications and problems that will be solvable based on SpecNet.

2.1 Methods

The key idea of SpecNet rests on the observation that feature maps, like most natural images, tend to have compact energy in the spectral domain. The compression can be achieved by retaining non-trivial values while zeroing out small entries. A threshold (β) can then be applied to configure the compression rate where larger β values result in more zeros in the spectral domain feature maps. Therefore, SpecNet represents a new design of the network architecture for convolution, tensor compression, and activation in the spectral domain and can be applied to both forward and backward propagation in network training and inference.

Compression in Feature Maps

Consider 2D-convolution with a stride of 1. In a standard convolutional layer, the output is computed by

$$y(i,j) = x * k = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n)\, k(i-m, j-n), \tag{2.1}$$

where x is an input matrix of size (M, N); k is the kernel with dimension (N_k, N_k), and $*$ indicates 2D convolution. The output y in the spatial domain has dimensions (M', N'), where $M' = M + N_k - 1$ and $N' = N + N_k - 1$. This process involves $\mathcal{O}(M'N'N_k^2)$ multiplications.

Convolution can be implemented more efficiently in the spectral domain as

$$Y = X \odot K, \tag{2.2}$$

where X is the transformed input in the spectral domain by FFT $X = \mathcal{F}(x)$, and K is the corresponding kernel in the spectral domain, $K = \mathcal{F}(k)$. $\odot$ represents element-by-element multiplication, which requires equal dimensions for X and

K. Therefore, x and k are zero-padded to match their dimensions (M', N'). Since there are various hardware optimizations for the FFT [Pratt et al. (2017)], it requires $\mathcal{O}(M'N' \log(M'N'))$ complex multiplications. The computational complexity of (2.2) is $\mathcal{O}(M'N')$ and so the overall complexity in the spectral domain is $\mathcal{O}(M'N' \log(M'N'))$. Depending on the size of the inputs and kernels, SpecNet can have a computational advantage over spatial convolution in some cases [Pratt et al. (2017)]. However, SpecNet is focused on reducing memory consumption for applications that are primarily limited by the available memory.

The compression of Y involves a configurable threshold β, which forces entries in Y with small absolute values (those less than β) to zero. This allows the thresholded map $\hat{Y}$ to be sparse and hence to store only the non-zero entries in $\hat{Y}$, thus saving memory.

The backward propagation step requires the calculation of the error δ_X for the previous layers, and the gradients Δ_K for k. Let δ_Y be the error from the next layer, and X_0, k_0 be the input and kernel of the convolutional layer stored in the forward propagation, respectively. Then

$$\begin{aligned} \delta_X &= \nabla_X L|_{X=X_0} = \delta_Y \odot K_0 \\ \Delta_K &= \nabla_K L|_{K=\mathcal{F}(k_0)} = \delta_Y \odot X_0, \end{aligned} \tag{2.3}$$

where L is the loss function. After obtaining its gradient in the spectral domain Δ_K, the IFFT is applied. Then the $N_k \times N_k$ matrix for the update of k can be expressed as

$$k_1 = k_0 + \lambda[\mathcal{F}^{-1}(\Delta_K)]_{N_k \times N_k}, \tag{2.4}$$

where λ is the learning rate. The kernels are updated after obtaining Δ_K, the gradient in the spectral domain, by using the inverse FFT and downsampling.

Note that after the gradient update of Δ_K, the kernel is further converted from the spectral domain back into the spatial domain using the inverse FFT to save kernel storage.

A more general case of 2D-convolution with arbitrary integer stride can be viewed as a combination of 2D-convolution with stride of 1 and uniform down-

sampling, which can also be implemented in the spectral domain [Pratt et al. (2017)].

Activation Function for Preserving the Symmetry Structure

In SpecNet, the activation function for the feature maps is designed to perform directly in the spectral domain. For each complex entry in the spectral feature map,

$$f(a + ib) = h(a) + ig(b) \tag{2.5}$$

where

$$h(x) = g(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{2.6}$$

The tanh function is used in (2.5) as a proof-of-concept design for this study. Other activation functions may also be used, but must fulfill the following:

1. They allow inexpensive gradient calculation.
2. Both $g(x)$ and $h(x)$ are monotonic nondecreasing
3. The functions are odd, i.e. $g(-x) = -g(x)$.

The first and second rules are standard requirements for nearly all popular activation functions used in modern CNN design. The third rule in SpecNet is applied to preserve the conjugate symmetry structure of the spectral feature maps so that they can be converted back into real spatial features without generating pseudo phases. The 2D FFT is

$$X(p, q) = \mathcal{F}(x) = \sum_{m=0}^{M+N_k-2} \sum_{n=0}^{N+N_k-2} w_M^{pm} w_N^{qn} x(m, n)$$

where $w_M = e^{-2\pi i/(M+N_k-1)}$, $w_N = e^{-2\pi i/(N+N_k-1)}$. If x is real, i.e., the conjugate

of x is itself ($\bar{x} = x$), and

$$\begin{aligned}
&X(M + N_k - 1 - p_0, N + N_k - 1 - q_0) \\
=&\sum_{m=0}^{M+N_k-2} \sum_{n=0}^{N+N_k-2} w_M^{(M+N_k-1-p_0)m} w_N^{(N+N_k-1-q_0)n} x(m, n) \\
=&\sum_{m=0}^{M+N_k-2} \sum_{n=0}^{N+N_k-2} w_M^{-p_0 m} w_N^{-q_0 n} x(m, n) \\
=&\overline{X(p_0, q_0)}
\end{aligned} \tag{2.7}$$

Therefore, $g(x)$ must be odd to retain the symmetry structure of the activation layer to ensure that

$$\begin{aligned}
f(\overline{a + ib}) &= h(a) + ig(-b) \\
&= h(a) - ig(b) = \overline{f(a + ib)}.
\end{aligned} \tag{2.8}$$

If the symmetry structure of δ_Y in (2.3) is also maintained, the gradients in the spatial domain will be real-valued after the inverse FFT in (2.4), and can be added to k_0 directly.

Let X_0 be the input to the activation layer in forward propagation, and δ_Y be the error from the next layer. The error for the previous layer in backward propagation can be calculated by

$$\begin{aligned}
\delta_X =&\{1 - [\tanh(\Re(X_0))]^2\} \odot \Re(\delta_Y) + \\
&i\{1 - [\tanh(\Im(X_0))]^2\} \odot \Im(\delta_Y).
\end{aligned} \tag{2.9}$$

Implementation Details

SpecNet stores the kernels in the spatial domain as $N_k \times N_k$ matrices. Therefore, given the input feature maps in the spectral domain, each kernel should be up-sampled to the size of the inputs by padding with zeros, and then transformed to

Algorithm 1 Forward propagation of the convolutional block in SpecNet

Input: feature maps x from the last layer with size of $M \times N$; kernel k ($N_k \times N_k$); threshold β.

if x in the spectral domain **then**
    Set $M' = M$, $N' = N$ and $X = x$.
else
    Set $M' = M + N_k - 1$ and $N' = N + N_k - 1$.
end if
for $i = 1$ to M' **do**
    for $j = 1$ to N' **do**
        if X is None **then**
            $\hat{x}(i,j) = x(i,j)$ if $i \leqslant M$ and $j \leqslant N$, and $\hat{x}(i,j) = 0$ otherwise.
        end if
        $\hat{k}(i,j) = k(i,j)$ if $i \leqslant N_k$ and $j \leqslant N_k$, and $\hat{k}(i,j) = 0$ otherwise.
        Calculate $K = \mathcal{F}(\hat{k})$ and $X = \mathcal{F}(\hat{x})$ if X is None.
    end for
end for
    Calculate Y after convolution according to (2.2).
    Obtain $\hat{Y}$ where $\hat{Y}(i,j) = Y(i,j)$ if $|Y(i,j)| > \beta$, and $\hat{Y}(i,j) = 0$ otherwise.
    Get $Z = f(\hat{Y})$ where f is defined in (2.5)

Output: The feature map in the spectral domain: Z.

the spectral domain with the FFT. The complete forward propagation of the convolutional block (including convolution and activation operations) in the spectral domain is shown in Algorithm 1.

The pooling methods in SpecNet are implemented in the spatial domain after transforming the activated frequency feature maps back into the spatial domain using the IFFT. Since the activation function preserves conjugate symmetry, the corresponding spatial feature maps are real valued and the same pooling operation (max pooling or average pooling) used in standard CNNs can be used seamlessly in SpecNet.

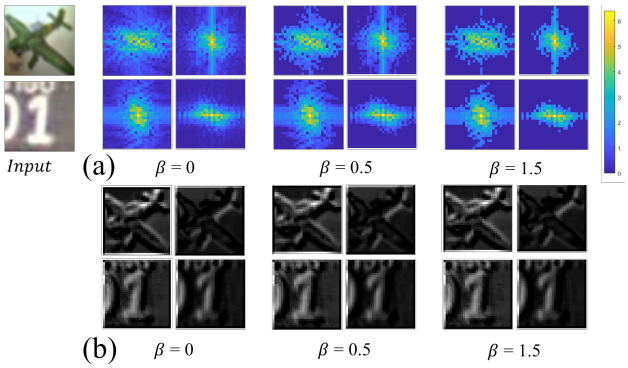

Figure 2.1: Feature maps of SpecDenseNet after the first convolutional layer for different inputs. (a) Feature maps after two different kernels in the spectral domain with three different thresholds β. (b) Feature maps, converted from the spectral domain to the spatial domain by utilizing the inverse Fourier transform with different thresholds β.

2.2 Experiments

We demonstrate the feasibility of SpecNet using three benchmark datasets: CIFAR, SVHN and ImageNet, and by comparing the performance of SpecNet implementations of several state-of-the-art networks (AlexNet, VGG16 and DenseNet) with their standard implementations. All the networks were trained by mini batch stochastic gradient descent (SGD) with a batch size of 64. The initial learning rate was set to 0.0001 and was reduced by half every 50 epochs. The momentum of the optimizer was set to 0.95 and a total of 300 epochs were trained to ensure convergence.

Datasets and Training

CIFAR-10 is a ten-class dataset of small colored natural images Krizhevsky and Hinton (2009). In our experiment, 50,000 images were used for training and 10,000 images were used for testing. All images of CIFAR 10 were resized to 32 by 32 pixels, and each channel was normalized with respect to its mean and standard deviation [Huang et al. (2017)].

SVHN is a dataset consisting of colored digit images with 32 by 32 pixels each

Table 2.1: Detailed network structure for DenseNet and SpecDenseNet

DenseNet			
Layers	Output size	Structure	
Input	32×32	Input	
Convolution	32×32	3×3 kernel, BN, ReLU	
Pooling	16×16	MaxPool (window size 2×2)	
Dense Block	16×16	1×1 kernel, BN, ReLU 3×3 kernel, BN, ReLU	×6
Classification Layer	1×1	GlobalAveragePool Fully-connected, SoftMax (10 classes)	
SpecDenseNet			
Layers	Output size	Structure	
Input	32×32	Input	
Convolutional Block	35×35	FFT	
	35×35	FConv2D (3×3 kernels), Activation	
	32×32	IFFT	
Pooling	16×16	MaxPool (window size 2×2)	
Dense Block	19×19	FFT	
	19×19	FConv2D (64 1×1 kernels), Activation FConv2D (64 3×3 kernels), Activation	×6
	16×16	IFFT	
Classification Layer	1×1	GlobalAveragePool Fully-connected, SoftMax (10 classes)	

image [Netzer et al. (2011)]. The dataset contains 99289 images: 73257 images for training and 26032 images for testing. Images were channel-normalized in mean and standard deviation.

ImageNet is ILSVRC 2012 classification dataset which contains more than 1.2 million images for training and 50000 images for testing. We apply data augmentation for images as in [Huang et al. (2017)], and cropped the image size to 224×224.

Results

Firstly, we empirically show the impact when two different thresholds ($\beta = 0.5$ and $\beta = 1.5$) are applied to the feature maps. The results are shown in Fig. 2.1. Observe that the feature maps in the spectral domain are compressed, but this does not significantly impact the feature maps in the spatial domain.

Further, we evaluated the proposed SpecNet using three widely used CNN

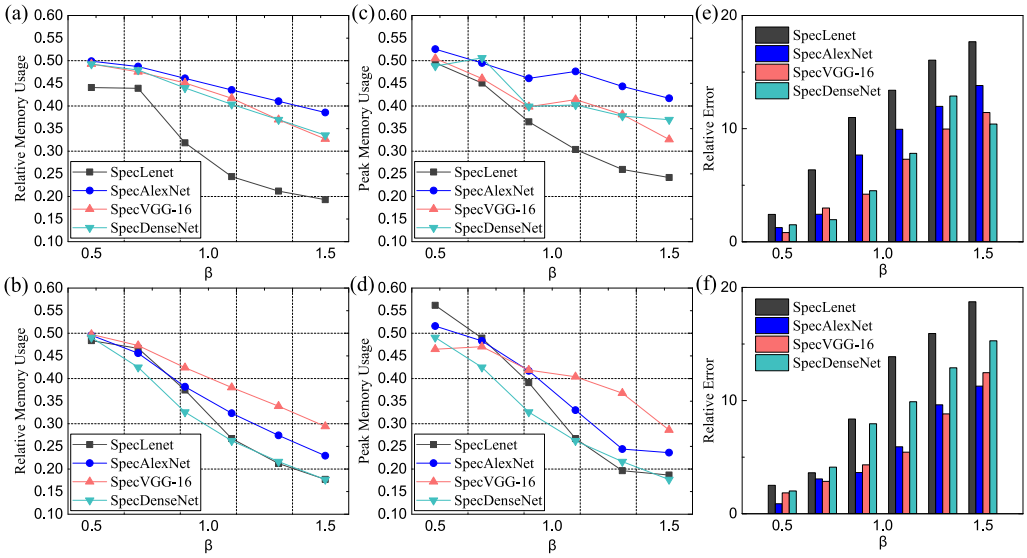

Figure 2.2: Memory consumption and testing performance of SpecNet compared with AlexNet, VGG, and DenseNet [LeCun et al. (1995); Krizhevsky et al. (2012a,b); Huang et al. (2017); Wang et al. (2017a)] on two datasets. To make the comparison fair, we retain the analogous structures which we call SpecAlexNet, SpecVGG and SpecDenseNet. (a)(b)(c) relative memory consumption and (d)(e)(f) relative error of SpecNets tested on CIFAR-10 and SVHN.

architectures including AlexNet [Krizhevsky et al. (2012a)], VGG [Krizhevsky et al. (2012b)] and DenseNet [Huang et al. (2017)]. We use the prefix 'Spec' to stand for the SpecNet implementation of each network. To ensure fair comparisons, the SpecNet networks used identical network hyper-parameters as the native spatial domain implementations. The experiments also use the same conditions for image preprocessing, parameter initialization, and optimization settings.

Architectures for DenseNet and SpecDenseNet are in Table 2.1 and the other three networks are detailed in the supplementary material. The experimental results on CIFAR-10 and SVHN are shown in Fig. 2.2.

Figures 2.2(a)(b) and (c)(d) compare the average memory usage and peak memory usage of the SpecNet implementations of three different networks over a range of beta values from 0.5 to 1.5. We compute relative memory consumption and accu-

Table 2.2: Memory consumption and testing performance of SpecNet compared with AlexNet, VGG, and DenseNet. All values are relative values comparing SpecNet with its implementation in spatial domain.

	Top1 Val. Error (%)	Top5 Val. Error (%)	Peak Memory (%)	Average Memory (%)
Spec-AlexNet	8.3	3.6	48.1	49.3
Spec-VGG16	7.2	2.3	42.4	46.7
Spec-DenseNet169	6.9	1.8	36.6	40.8

racy by: memory (accuracy) of SpecNet / the memory (accuracy) in the original implementations. When compared with their original models, all SpecNet implementations of the three networks can save at least 50% memory with negligible loss of accuracy, indicating the feasibility of compressing feature maps within the SpecNet framework. With increasing β value, all models show reduction in both average and peak memory usage. The rates of memory reduction are different between different network architectures, which is likely caused by different feature representations in the various network designs.

We also tested SpecNet for the three different networks on the ImageNet dataset with the β value set to 1.0. As shown in Table. 2.2, both the average and the peak memory consumption of SpecAlexNet, SpecVGG, and SpecDenseNet are less than half of the original implementations. The peak memory usage of SpecNet is reduced even more, which is due to the specifics of the implementation of convolution in Tensorflow.

Figures 2.2(e)(f) compare the relative accuracy of the SpecNet implementations of the three different networks over a range of β values from 0.5 to 1.5. While SpecNet typically compresses the models, there is a penalty in the form of increased error in comparison to the original model with full spatial feature maps. The average accuracy of SpecAlexNet, SpecVGG, and SpecDenseNet can be higher than 95% when β is smaller than 1.0. Table 2.2 shows the testing accuracy on ImageNet. Compared with implementation in spatial domain, SpecNet has a slight decrease in the top-1 accuracy but almost the same top-5 accuracy.

Fig. 2.3 shows the training curve of the SpecNet implementations of three different networks with their implementations in spatial domain. From the training

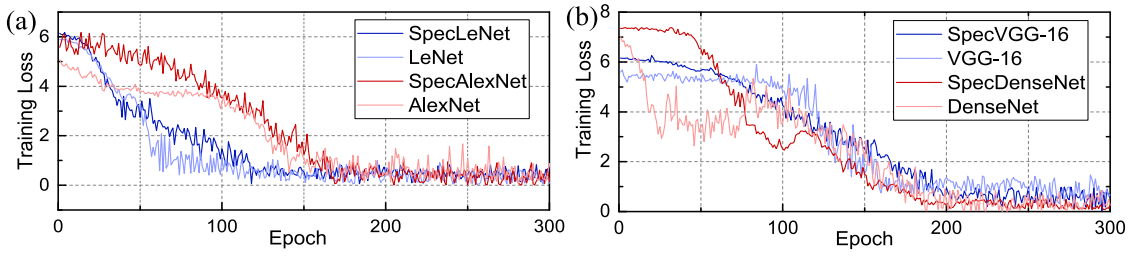

Figure 2.3: Training curves of SpecNet comparing with AlexNet, VGG-16 and DenseNet on CIFAR-10 dataset.

Table 2.3: Comparison of relative memory usage for different memory efficient implementations applied to VGG and DenseNet. All the methods are tested on CIFAR-10.

Model	VGG-16		DenseNet	
	Memory	Accuray	Memory	Accuracy
INPLACE-ABN [Rota BulÃ² et al. (2018)]	0.521	0.914	0.580	0.929
Chen Meng et al. [Meng et al. (2017)]	0.656	0.921	0.553	0.932
Efficient-DenseNets [Pleiss et al. (2017)]	N/A	N/A	0.443	0.933
Nonuniform Quantization [Sun et al. (2016)]	0.8	0.917	0.771	0.922
LQ-Net [Zhang et al. (2018)]	0.676	0.919	0.644	0.923
HarDNet [Chao et al. (2019)]	0.463	0.921	0.442	0.933
SpecNet	0.370	0.918	0.370	0.925

curve, the SpecNet implementations converge with similar rates as the implementations in the spatial domain. Section 2.1 also showed that the computation speed is related to the size of the inputs and kernels, and SpecNet is advantageous in some cases. Therefore, training speed of SpecNet is comparable with the network when implemented in the spatial domain, but with the added benefit of memory efficiency.

Table 2.3 shows a comparison between SpecNet and other recently published memory-efficient algorithms. The experiments investigate memory usage when training VGG and DenseNet on the CIFAR-10 dataset. SpecNet outperformed all the listed algorithms and resulted in the lowest memory usage while maintaining high testing accuracy. It is notable that SpecNet is independent of the methods listed in the table, and these techniques may be applied along with SpecNet to further reduce memory consumption.

2.3 Conclusion

We have introduced a new CNN architecture called SpecNet, which performs both the convolution and activation operations in the spectral domain. We evaluated SpecNet on two competitive object recognition benchmarks, and demonstrated the performance with three state-of-the-art algorithms to show the efficacy and efficiency of memory reduction. In some cases, SpecNet can reduce memory consumption by 63% without significant loss of performance. It is also notable that SpecNet is only focused on the sparse storage of feature maps. In the future, it should be possible to merge other methods, such as model compression and scheduling, with SpecNet to further improve memory usage.

3 FULLY AUTOMATED DIAGNOSIS OF ANTERIOR CRUCIATE LIGAMENT TEARS ON KNEE MR IMAGES BY USING SPECNET

3.1 Introduction

Deep learning has been successfully used for various medical imaging applications including image reconstruction [Liu (2018); Richardson and Petscavage (2011); Zhu et al. (2018)], tissue and lesion segmentation [Liu et al. (2017); Norman et al. (2018); Zhou et al. (2018)], and disease detection and characterization [Liu et al. (2018); Lakhani and Sundaram (2017); De Fauw et al. (2018); Bien et al. (2018); Arevalo et al. (2016)]. The use of deep learning for disease detection would be especially important for clinical radiologists as it could maximize diagnostic performance while reducing subjectivity and errors due to distraction and fatigue. Most prior work on the use of deep learning for disease detection has focused on identifying abnormalities such as masses on breast mammograms [Arevalo et al. (2016); Kooi et al. (2017)], lung nodules, opacities, and cardiomegaly on chest radiographs [Lakhani and Sundaram (2017); Ciompi et al. (2015)], and lung nodules and interstitial lung disease on chest computed tomography [Ciompi et al. (2015); Anthimopoulos et al. (2016)].

However, applications of deep learning to disease detection on magnetic resonance imaging (MRI) is particularly challenging as it often requires analyzing complex abnormalities on multiple slices of different image datasets [Liu et al. (2018); Pedoia et al. (2018); Bien et al. (2018)]. MRI data is usually much larger than natural image data, and so applying CNNs to MRI requires increased memory, often beyond what is available in modern GPUs. Previous work includes attempts to solve this by simplifying the network structure [Wen et al. (2016); Li et al. (2016)] and by model compression [Wu et al. (2016); Courbariaux and Bengio (2016); Hanson and Pratt (1989); Denton et al. (2014)]. However, these methods often sacrifice accuracy, may not satisfy the requirements for medical implementation.

Anterior cruciate ligament (ACL) tears are a common sports-related musculoskeletal injury [Gianotti et al. (2009)]. Detecting an ACL tear relies on evaluating

an obliquely oriented structure on multiple image slices with different tissue contrasts using a combination of MRI features including fiber discontinuity, changes in contour, and signal abnormality within the injured ligament [Lee et al. (1988)]. Investigation of the ability of a deep learning approach to detect an ACL tear would be useful to determine whether deep learning could aid in the diagnosis of complex musculoskeletal abnormalities on MRI. We have developed a novel fully-automated system for detecting an ACL tear utilizing two deep convolutional neural networks (CNN) to isolate the ACL on MR images followed by a classification CNN to detect structural abnormalities within the isolated ligament. The networks are implemented by our SpecNet platform which makes its training very memory efficient. This study was performed to investigate the feasibility of using the deep learning-based approach to detect a full-thickness ACL tear within the knee joint on MRI using arthroscopy as the reference standard. This study provides a good example of the application of SpecNet to the solution of a newly developed network.

3.2 Methods

Fully-Automated ACL Tear Detection System

The proposed deep learning-based ACL tear detection system consisted of three separate CNNs. The first CNN selected the image slices that contained the ACL from the entire MR image dataset. The second CNN isolated the region of the intercondylar notch that contained the ACL on the selected image slices to narrow the range of information used for subsequent image recognition. The third classification CNN evaluated the isolated ACL on the selected image slices to determine the presence or absence of an ACL tear. The three CNNs were connected in a cascaded fashion to create a fully-automated processing pipeline as shown in Figure 3.1. The processing pipeline framework was implemented by SpecNet platform and make it possible to train on our hardware. The computing environment involves Python (version 3.7, Python Software Foundation, Wilmington, DE) and Matlab (version 2018a, MathWorks, Natick, MA). The CNNs were coded using TensorFlow (version

1.12, Google, Mountain View, CA).

The first slice selection CNN was adapted from LeNet-5, which was originally proposed for natural image analysis [LeCun et al. (2015)] but recently used for many medical imaging applications [Malon and Cosatto (2013); Sarraf et al. (2016)]. The CNN, shown in the top of Figure 3.1, had two sets of convolutional layers, which were followed by two fully-connected layers and then a SoftMax classifier. The CNN was modified to allow an input image size of 448 Ã— 448 with the images normalized to a range of 0 and 1 with respect to the maximum MRI signal. Two output classes were defined, one class for image slices that contained the ACL and the other for image slices that did not contain the ACL.

The second ligament isolation CNN was adapted from You Only Look Once (YOLO), which has demonstrated top performance in many image object detection challenges [Redmon et al. (2016); Huang et al. (2017)]. The CNN, shown in the middle of Figure 3.1, had 24 convolutional layers followed by two fully connected layers. The convolutional layers were used to extract useful image features from the selected image slices that contained the ACL. The fully connected layers then used the extracted image features to predict the coordinates of a rectangle box that defined the region of the intercondylar notch that contained the ACL. The cropped images of the pre-defined rectangle box that contained the ACL on the selected image slices were downsized to 112×112 and used as input images to the classification CNN.

The classification CNN was adapted from Densely Connected Convolutional Networks (DenseNet) [Huang et al. (2017)], which has demonstrated top performance for image recognition in the Large Scale Visualize Recognition Challenge (LSVRC) image dataset [Simonyan and Zisserman (2014); Russakovsky et al. (2015)]. The CNN, shown in the bottom of Figure 3.1, contained three dense blocks, each of which was connected by a convolutional layer and a maxpooling layer. Dense connectivity was applied to enable efficient information sharing across the layers, which substantially reduced the network parameters needed for high throughput training. For the output, a final probability score for the presence or absence of an ACL tear on the cropped images that contained the ACL was obtained from a global

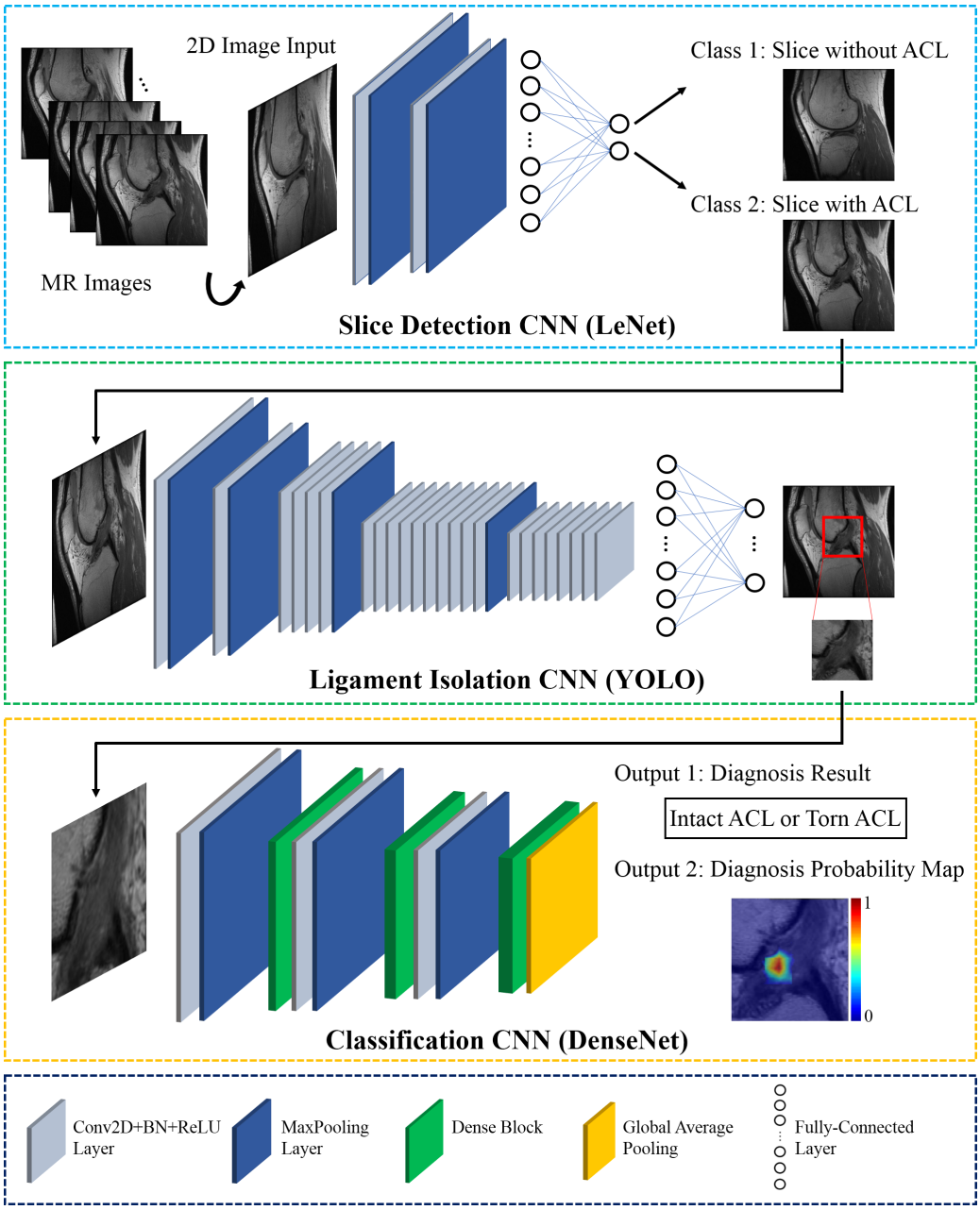

Figure 3.1: Illustration of the CNN architecture for the deep learning-based ACL tear detection system. The proposed method consisted of three separate CNNs connected in a cascaded fashion to create a fully automated image processing pipeline.

average pooling (GAP) layer which followed the dense blocks. The classification result was computed by the mean of the classification probability of an ACL tear on all image slices that contained the ACL. The GAP layer was further modified using a gradient back-propagation method [Zhou et al. (2016)] to calculate a diagnosis probability map matching the input image size, which showed the pixel-by-pixel probability of the presence of an ACL tear. To compare with other popular CNNs used in previous image classification applications [Liu et al. (2018); Lakhani and Sundaram (2017)], two additional classification CNNs adapted from the commonly used image recognition networks Very Deep Convolutional Networks for Large-Scale Image Recognition (VGG16) [Simonyan and Zisserman (2014)] and AlexNet [Krizhevsky et al. (2012a)] were also evaluated and compared with the proposed CNN adapted from DenseNet.

MR Image Datasets

The retrospective study was performed in compliance with the Health Insurance Portability and Accountability Act (HIPAA) regulations, with approval from our Institutional Review Board, and with a waiver of informed consent. MR image datasets were obtained from 175 subjects with a surgically confirmed ACL tear (98 men and 77 women with an average age of 27.5 years and with an age range between 16 years and 47 years) and 175 subjects with a surgically confirmed intact ACL (100 men and 75 women with an average age of 39.4 years and with an age range between 17 years and 51 years). All subjects underwent an MRI examination of the knee and subsequent arthroscopic knee surgery at our institution between December 15, 2010 and October 15, 2017.

All MRI examinations were performed on the same 3T scanner (Signa Excite HDx, GE Healthcare, Waukesha, WI) using an 8-channel phased-array extremity coil (Invivo, Orlando, FL) and consisted of a standard axial fat-suppressed T2-weighted fast spin-echo sequence, standard sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo sequences, and standard coronal proton density-weighted and fat-suppressed proton density-weighted fast spin-echo

sequences. Subjects with and without an ACL tear were selected by a fellowship trained musculoskeletal radiologists with 15 years of clinical experience using a medical record achieving system (Hyperspace, Epic, Verona, WI). The radiologist reviewed the schedule of arthroscopic knee surgeries performed at our institution and determined whether the subject undergoing arthroscopy had an MRI examination of the same knee performed on the same 3T scanner within four months prior to surgery.

3.3 Training and Evaluation of the ACL Tear Detection System

Training and evaluation of the ACL tear detection system were performed on a desktop computer running a 64-bit Linux operating system (Ubuntu 16.04) with an Intel Core i7-7700K CPU with 32 GB DDR4 RAM and two Nvidia GeForce GTX 1080 graphic cards (Nvidia driver 384.130) with 2560 CUDA cores and 8GB GDDR5 RAM . Training of the system was performed for each CNN individually as described in the Appendix. However, once the training was completed for all three CNNs, the ACL tear detection system operated as a fully-automated end-to-end network. To train the slice selection CNN and ligament isolation CNN, a fellowship-trained musculoskeletal radiologist with 15 years of clinical experience manually identified the MR image slices that contained the ACL and then manually outlined the region of the intercondylar notch that contained the ACL on the selected MR image slices. However, once the training was completed, the ACL tear detection system provided fully automated isolation of the ACL on all MR image slices to be used as inputs into the classification CNN.

Training the slice selection CNN was performed using the sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo images acquired during the MRI examination. The reference standard for training the slice selection CNN was manual identification of the MR image slices that contained the ACL performed by a fellowship-trained musculoskeletal radiologist with 15 years of clinical

experience. The radiologist evaluated the sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo images side-by-side using customized software developed in MATLAB (version 2016a, MathWorks, Natick, MA) and provided a label for each MR image slice with the following values: 0=slice that contained the ACL and 1=slice that did not contain the ACL. This resulted in 832 slices that contained the ACL from a total of 12258 slices in the training dataset, 199 slices that contained the ACL from a total of 2986 slices in the validation dataset, and 449 slices that contained the ACL from a total of 6381 slices in the hold-out test dataset. To increase training efficiency, image augmentation was performed using random geometrical operations including image translation, rotation, and flipping [Krizhevsky et al. (2012a)]. A dropout technique with a dropout rate of 20% between convolutional layers was also applied to reduce training overfitting [Hinton et al. (2012)]. The sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo images in DICOM format were resized to 448Ã—448 matrix size, normalized between 0 and 1 with respect to the maximum MRI signal, and convert to TFRecord file format, Tensorflow's binary storage format. The network was trained at a batch size of 16 images using multi-class cross-entropy loss [Long et al. (2014)] and was updated using Stochastic Gradient Descent [Bottou (2010)] at a fixed learning rate of 0.01 and momentum of 0.9 for a total of 20 epochs of the training data.

Training the ligament isolation CNN was performed using the selected sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo image slices from the slice detection CNN. The reference standard for training the ACL isolation CNN was manual outline of the region of the intercondylar notch that contained the ACL performed by a fellowship-trained musculoskeletal radiologist with 15 years of clinical experience. The radiologist evaluated the selected sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo image slices using the same customized software and placed a rectangle box around the ACL. The x-y coordinate information of the top-left and bottom-right corners of the rectangle box was labeled and used for network training. The CNN was trained to automatically output the coordinate information of the rectangular box outlining the

region of the intercondylar notch that contained the ACL. An automated cropping process was performed to extract the image patches that contained the isolated ACL for use as input images into the classification CNN. To address the challenge for training the large YOLO network, the weights of the network were initialized by the pre-trained model of ImageNet [Russakovsky et al. (2015)] and retrained using our dataset. The network was trained at a batch size of 16 images using multi-parts loss [Redmon et al. (2016)] and was updated using an adaptive gradient-based optimization algorithm [Kingma and Ba (2014)] with an initial learning rate of 0.001 for a total of 100 epochs in the training data.

Training the classification CNN was performed using the cropped sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo image patches that contained the ACL from the ligament isolation CNN. The reference standard for training the classification CNN was the presence or absence of an ACL tear at arthroscopic knee surgery. A label was provided for each extracted image patch that contained the isolated ACL on each selected sagittal proton density-weighted and fat-suppressed T2-weighted fast spin-echo image slices with the following values: 0=ACL torn and 1=ACL intact. This resulted in 401 image patches with an ACL tear from a total of 832 slices in the training dataset, 99 image patches with an ACL tear from a total of 199 slices in the validation dataset, and 216 image patches containing an ACL tear from a total of 449 slices in the hold-out test subjects. To increase the training efficiency for DenseNet, the weights of the network were initialized by the pre-trained model of ImageNet [Huang et al. (2017)] and then retrained on our dataset using a memory efficient training strategy Pleiss et al. (2017). The network was trained at a batch size of 32 image patches and was updated using an adaptive gradient-based optimization algorithm [Kingma and Ba (2014)] with an initial learning rate of 0.001 for a total of 170 epochs in the training data. The classification CNNs adapted from DenseNet, VGG16, and AlexNet were individually trained and evaluated.

3.4 Evaluation of Clinical Radiologists

To compare the diagnostic performance of the ACL tear detection system with clinical radiologists, a fellowship-trained musculoskeletal radiologist with 13 years of clinical experience, musculoskeletal radiology fellow, second year radiology resident, third year radiology resident, and fourth year radiology resident independently reviewed the MRI examinations of all 100 subjects in the hold-out test dataset using a Picture Archiving and Communication System (PACS) work-station (McKesson Corporation, San Francisco, CA). The clinical radiologists, who were blinded to the findings of the arthroscopic knee surgeries, used all five sequences in the MRI examination together to determine the presence or absence of an ACL tear. The radiologists received no formal training or calibration session prior to reviewing the MRI examinations.

3.5 Statistical Analysis

Statistical analysis was performed using MATLAB (version 2013a, MathWorks, Natick, MA) and MedCalc (version 14.8; MedCalc Software, Ostend, Belgium) with statistically significance defined as a p-value less than 0.05. Contingency tables and sensitivity and specificity for determining the presence or absence of an ACL tear using arthroscopy as the reference standard for the hold-out test dataset were determined for the musculoskeletal radiologist, musculoskeletal radiology fellow, radiology residents, proposed ACL tear detection system with the classification CNN adapted from DenseNet, and alternative ACL tear detection systems with the classification CNNs adapted from VGG16 and AlexNet. Receiver operating characteristic (ROC) analysis was used to further evaluate the diagnostic performance of the proposed and alternative ACL tear detection systems with area under the curves (AUCs) compared using a nonparametric approach [DeLong et al. (1988)]. The Youden index was used to determine the optimal sensitivity and specificity of the proposed and alternative ACL tear detection systems [Fluss et al. (2005)]. Two-sided exact binomial tests were used to calculate 95% confidence intervals

for the sensitivity and specificity of the proposed and alternative ACL tear detection systems and the clinical radiologists and for the AUCs of the proposed and alternative ACL tear detection systems. Statistically significant differences between the sensitivity and specificity of the proposed ACL tear detection system and the clinical radiologists was defined as the sensitivity and specificity point estimates of the clinical radiologists located outside the 95% confidence intervals of the AUC for the machine [De Fauw et al. (2018)].

3.6 Results

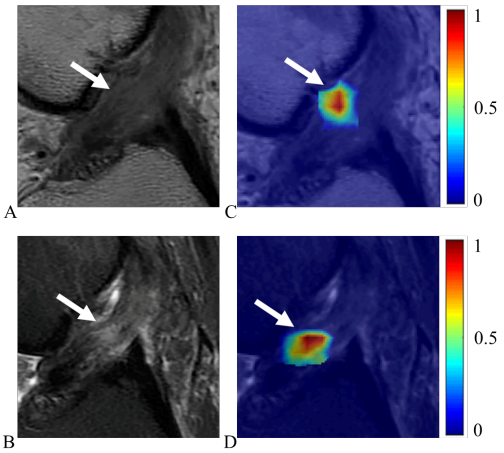

Figure 3.2: Images in a 28-year-old man with a surgically confirmed ACL tear for which the ACL tear detection system interpreted as the presence of an ACL tear. (A) Cropped sagittal proton density weighted fast spin-echo image and (B) cropped sagittal fat-suppressed T2-weighted fast spin-echo image of the knee analyzed by the classification CNN show disruption of fibers and increased signal within the ACL (arrows). (C) Probability map for the proton density weighted fast spin-echo image and (D) probability map for the sagittal fat-suppressed T2-weighted fast spin-echo image show the high probability areas in the ligament on which the machine based its interpretation of an ACL tear (arrows).

The training times for the slice selection, ligament isolation, and classification CNNs were 0.82 hours, 5.11 hours, and 5.70 hours respectively in the training

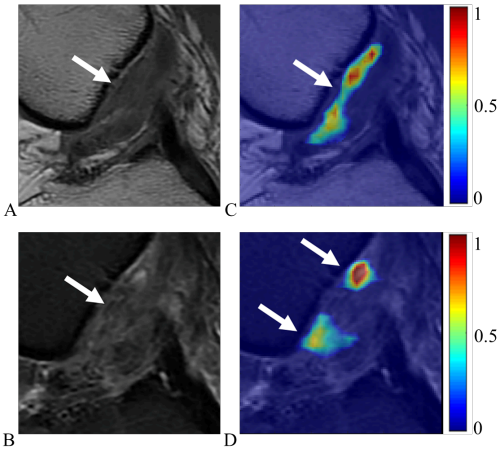

Figure 3.3: Images in a 26-year-old man with a surgically confirmed anterior cruciate ligament (ACL) tear that the ACL tear detection system interpreted as present. (a) Cropped sagittal proton densityâŁ“weighted fast spin-echo MR image and (b) cropped sagittal fat-suppressed T2-weighted fast spin-echo MR image of the knee analyzed by the classification convolutional neural network show disruption of fibers and increased signal within the ACL (arrow). (c) Probability map for the proton densityâŁ“weighted fast spin-echo image and (d) probability map for the sagittal fat-suppressed T2-weighted fast spin-echo image show the high-probability areas in the ligament on which the machine based its interpretation of an ACL tear (arrows).

datasets. However, the average time for the ACL tear detection system to provide an interpretation of the presence or absence of an ACL tear for one subject was only 9 seconds using the trained networks.

Table 3.1 compares the sensitivity, specificity, and AUC for determining the presence or absence of a surgically confirmed ACL tear for the proposed ACL tear detection system with the classification CNN adapted from DenseNet and the alternative ACL tear detection systems with the classification CNNs adapted from VGG16 and AlexNet. All classification CNNs performed well with point estimates of the sensitivity and specificity ranging between 0.89 and 0.96 and between 0.88 and 0.96 respectively and AUC ranging between 0.90 and 0.98. However, the proposed ACL tear detection system with the classification CNN adapted from DenseNet

Table 3.1: Sensitivity, specificity, and AUC for the proposed ACL tear detection system with the classification CNN adapted from DenseNet and the alternative ACL tear detection systems with the classification CNNs adapted from VGG16, and AlexNet for determining the presence or absence of an ACL tear on the hold-out test dataset.

CNN	Sensitivity (95% CI)	Specificity (95% CI)	AUC (95% CI)
DenseNet	0.96 (0.89 to 1)	0.96 (0.86 to 1)	0.98 (0.93 to 1)
VGG16	0.92 (0.82 to 0.98)	0.92 (0.82 to 0.98)	0.95 (0.90 to 0.99)
AlexNet	0.89 (0.76 to 0.96)	0.88 (0.76 to 0.96)	0.9 (0.83 to 0.96)

Table 3.2: Sensitivity and specificity for the musculoskeletal radiologist, musculoskeletal radiology fellow, radiology residents, and proposed ACL tear detection system for determining the presence or absence of an ACL tear on the hold-out test dataset.

	Sensitivity (95% CI)	Specificity (95% CI)	AUC (95% CI)
Radiologist	0.98 (0.82 to 1)	0.98 (0.88 to 1)	0.98 (0.95 to 1)
Fellow	0.96 (0.84 to 1)	0.98 (0.88 to 1)	0.98 (0.95 to 1)
Resident 1	0.96 (0.78 to 0.96)	0.9 (0.76 to 0.96)	0.97 (0.94 to 1)
Resident 2	0.98 (0.86 to 1)	0.98 (0.88 to 1)	0.93 (0.88 to 0.98)
Resident 3	0.98 (0.88 to 1)	0.98 (0.88 to 1)	0.97 (0.94 to 1)
Machine	0.96 (0.89 to 1)	0.96 (0.86 to 1)	0.98 (0.93 to 1)

had the highest overall diagnostic performance for detecting an ACL tear.

Tables 3.2 shows the contingency tables and sensitivity and specificity respectively for the musculoskeletal radiologist, musculoskeletal radiology fellow, radiology residents, and proposed ACL tear detection system for determining the presence or absence of a surgically confirmed ACL tear. The point estimates of the sensitivity and specificity of the proposed ACL tear detection system at the optimal threshold of the Youden index was 0.96 and 0.96 respectively. In comparison,

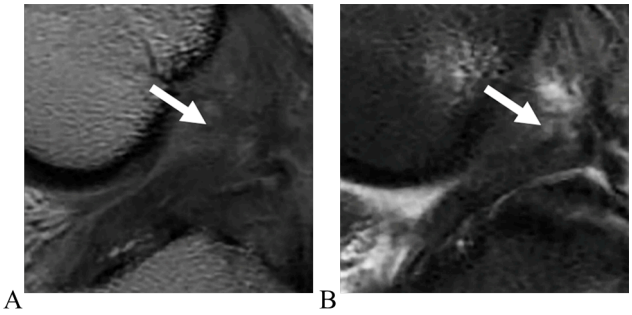

Figure 3.4: Twenty-year-old year old male with a surgically confirmed ACL tear for which the ACL tear detection system interpreted as an intact ACL. (A) Cropped sagittal proton density weighted fast spin-echo image and (B) cropped sagittal fat-suppressed T2-weighted fast spin-echo image of the knee analyzed by the classification CNN show fiber disruption and increased signal within the torn ACL (arrows).

the point estimates of the sensitivity of the clinical radiologists ranged between 0.96 and 0.98, while the point estimates of the specificity ranged between 0.90 and 0.98. In particular, the point estimates of both the sensitivity and specificity of the fellowship-trained musculoskeletal radiologist was 0.98. The proposed ACL tear detection system could create dense probability maps which showed the pixel-by-pixel probability of the presence of an ACL tear (Figures 3.3 and 3.3). For all 100 subjects in the hold-out test dataset, the machine had only two false negative interpretations of the absence of an ACL tear (Figure 3.4) and two false positive interpretations of the presence of an ACL tear (Figures 3.5).

Figure 3.6 shows the ROC curve describing the diagnostic performance of the proposed ACL tear detection system for determining the presence and absence of a surgically confirmed ACL tear. The AUC for the ACL tear detection system was 0.98 (95% CI of 0.93 to 1.000, $p<0.001$). For comparison, the point estimates representing the sensitivity and specificity of the musculoskeletal radiologist, musculoskeletal radiology fellows, and radiology residents for determining the presence and absence of an ACL tear were plotted on the figure and were in close proximity to the ROC curve of the ACL tear detection system. All sensitivity and specificity point

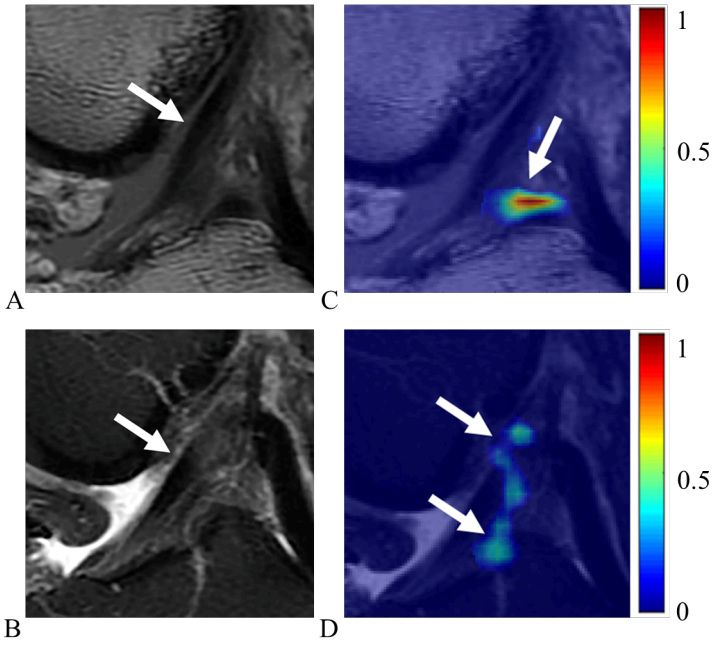

Figure 3.5: Twenty-one-year-old male with a surgically confirmed intact ACL for which the ACL tear detection system interpreted as an ACL tear. (A) Cropped sagittal proton density-weighted fast spin-echo image and (B) cropped sagittal fat-suppressed T2-weighted fast spin-echo image of the knee analyzed by the classification CNN show continuous fibers and normal signal within the intact ACL (arrows). (C) Probability map for the proton density weighted fast spin-echo image and (D) probability map for the sagittal fat-suppressed T2-weighted fast spin-echo image show the high probability areas in the ligament on which the machine based its interpretation of an ACL tear (arrows).

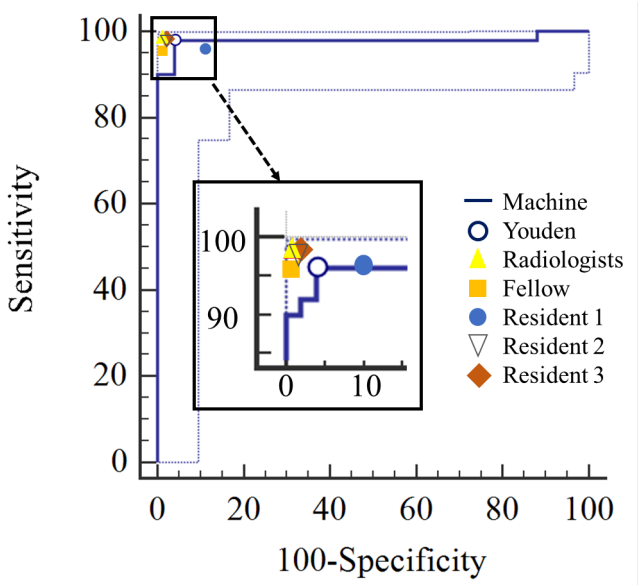

Figure 3.6: ROC curves describing the diagnostic performance of the proposed ACL tear detection system for determining the presence and absence of a surgically confirmed ACL tear. The AUC of the machine was 0.98, indicating high overall diagnostic accuracy. Sensitivity and specificity for the musculoskeletal radiologist, musculoskeletal radiology fellow, radiology residents, and machine at the optimal threshold of the Youden index are also plotted. Note that the sensitivity and specificity of the clinical radiologists are in close proximity to the ROC curve of the ACL tear detection system.

estimates of the clinical radiologists were located inside the 95% confidence intervals of the AUC for the proposed ACL tear detection system, indicating no statistically significant differences in diagnostic performance between the clinical radiologists and machine.

3.7 Conclusion

Our study has demonstrated the feasibility of using a deep learning-based approach to detect a full-thickness ACL tear within the knee joint on MRI using arthroscopy as the reference standard. This works also shows the feasibility of using SpecNet to de-

velop deep learning methods for MRI. By using memory efficient network, the ACL tear detection system still achieved high diagnostic performance for determining the presence or absence of an ACL tear with an AUC of 0.96.

4 DEEP LEARNING RISK ASSESSMENT MODELS FOR PREDICTING PROGRESSION OF RADIOGRAPHIC MEDIAL JOINT SPACE LOSS

4.1 Introduction

Osteoarthritis (OA) is one of the most prevalent and disabling chronic diseases in the United States and worldwide [Felson and Zhang (1998); Felson (2004)]. The knee is the joint most commonly affected by OA [Bedson et al. (2005); Peat et al. (2001)]. Identifying individuals at high risk for knee OA incidence and progression would provide a window of opportunity for disease modification during the earliest stages of the disease process when interventions such as weight loss, physical activity, and range of motion and strengthening exercises are likely to be most effective [Felson and Hodgson (2014)]. Identifying individuals at high risk for OA progression would also be useful for selecting the best subjects for inclusion in clinical trials investigating new disease modifying drugs [Karsdal et al. (2016); Hunter (2009)]. Clinical drug trials currently require large numbers of subjects and long follow-up periods due to the inherently different rates of disease progression in individuals with knee OA [Cicuttini et al. (2004); Wluka et al. (2002); Hanna et al. (2005); Raynauld et al. (2005); Gandy et al. (2002); Wirth et al. (2009)]. However, exclusive selection of subjects at high risk for knee OA progression for inclusion in clinical trials could reduce study size and duration, decrease the required financial resources, and potentially increase the likelihood of successful development of new disease modifying drugs [Karsdal et al. (2016); Hunter (2009)].

There is an important need to create OA risk assessment models for widespread use in clinical practice and clinical drug trials. However, current models, which have primarily used clinical and radiographic risk factors including age, gender, race, body mass index (BMI), history of knee injury, and Kellgren-Lawrence (KL) radiographic grade, have shown only moderate success for predicting the incidence [Yoo et al. (2016); Lazzarini et al. (2017); Kerkhof et al. (2014); Zhang et al. (2011)] and progression [Halilaj et al. (2018); LaValley et al. (2017)] of knee OA. Incorporation

of semi-quantitative and quantitative measures of knee joint pathology on baseline X-rays [Woloszynski et al. (2012); Janvier et al. (2017); Kraus et al. (2009); Janvier et al. (2017) and magnetic resonance (MR) images Joseph et al. (2018)] has improved the diagnostic performance of OA risk assessment models. However, the time and expertise needed to acquire these imaging parameters would make it impossible to incorporate them into widespread, cost-effective OA risk assessment models. Thus, new and improved strategies are needed to create comprehensive risk assessment models for the incidence and progression of knee OA.

Deep learning (DL) is an advanced artificial intelligence method which uses multiple levels of representation obtained by composing simple nonlinear modules that each transform the representation at one level into a representation at a higher and more abstract level [LeCun et al. (2015); Liu et al. (2019); Guan et al. (2019); Liu (2019)]. With the combination of enough such transformations, very complex features can be learned [Suzuki (2017)]. DL has tremendous potential for creating OA risk assessment models by providing a new rapid and fully-automated method to extract useful prognostic information from imaging studies. DL could potentially learn a representative subset of features on baseline imaging studies associated with the incidence and progression of knee OA. However, due to memory limitations, large scale DL models for OA predictions is challenging. Several previous attempts to apply machine learning models do not achieve significant diagnostic performance.

Our study was performed using a SpecNet architecture to develop and evaluate DL risk assessment models for predicting the progression of radiographic medial joint space loss using baseline knee X-rays. We hypothesize that DL models would have higher diagnostic performance for predicting the progression of radiographic joint space loss than traditional models using demographic and radiographic risk factors.

4.2 Methods

Eligible Study Participants

Eligible study participants were selected from the Osteoarthritis Initiative (OAI) database, a multi-center study which collected longitudinal clinical and imaging data over a nine-year follow-up period in 4796 subjects between the ages of 45 and 75 years with or at high risk for knee OA [Lester (2008)]. Study participants were selected from both the incidence cohort of subjects without radiographic knee OA but with knee pain and risk factors for OA incidence and the progression cohort of subjects with radiographic knee OA and risk factors for OA progression. The OAI was approved by the Committee on Human Research and the Internal Review Boards at University of California at San Francisco and at each individual clinical recruitment site.

Eligible study participants had the following clinical and imaging data publicly available in the OAI database: 1) age, gender, race, BMI, and history of knee injury (defined according to a question on the standardized OAI questionnaire as a non-specific acute injury preventing weight bearing for at least two days) at baseline, 2) KL grade of knee OA [Smith et al. (1957)] provided by central reading at baseline, 3) anatomic axis alignment (tibiofemoral angle) measurements [Felson et al. (2009)] provided by central reading at baseline, and 4) minimum medial joint space width measurements [Neumann et al. (2009)] provided by central reading at baseline and 48-month follow-up. KL grade, tibiofemoral angle, and minimum medial joint space width measurements were obtained from bilateral standing posterior-anterior knee X-rays acquired using standardized technique with a SynaFlexor fixed-flexion positioner [Kothari et al. (2004)]. There were 2301 subjects with 4602 knees with the above-mentioned clinical and imaging data available in the OAI database. One-hundred fifty-five knees in 154 subjects had a KL grade of 4 at baseline and were excluded as their minimum medial joint space width measurements would be expected to be 0mm at baseline. Thus, there was 2300 subjects with 4447 knees in the OAI database eligible to participate in our study.

Outcome Measure for the OA Risk Assessment Models

The outcome measure for the OA risk assessment models was a definitive progression of medial joint space loss on longitudinal bilateral standing posterior-anterior knee X-rays between baseline and 48-month follow-up measured using semi-automated software. The software determined the minimum joint space width across the medial compartment of the knee joint [Neumann et al. (2009)]. Definitive progression of radiographic joint space loss was defined according to the National Institute of Health OA Biomarkers Consortium Project as a greater than or equal to 0.7mm decrease in minimum medial joint space width measurements obtained between baseline and 48-month follow-up. This cutoff was based on the mean and standard deviation of one year changes in minimum medial joint space width measurements on bilateral standing posterior-anterior X-rays in 90 knees in the OAI reference control cohort with a KL grade of 0 and WOMAC pain score of 0 at both baseline and 24-month follow-up.

OA Risk Assessment Models

Traditional Risk Assessment Models

Traditional risk assessment models were developed using five alternative approaches including Random forest [Huang et al. (2016)], logistic regression [Yuan and Ghosh (2008)], and three different artificial neural networks (ANNs) [Lazzarini et al. (2017); Yoo et al. (2016); Kerkhof et al. (2014); Joseph et al. (2018); Amato et al. (2013); Hafezi-Nejad et al. (2017)]. Random forest is an ensemble-learning model that creates a multitude of decision trees during training with the output being the mode of the classifications of the individual trees. Logistic regression is a multivariable method for modeling binary classification that uses a logistic function to analyze the input to provide a confidence score between 0 and 1 for the output. The first ANN model (ANN), illustrated in Figure 4.1, had an identical architecture as an ANN that showed high diagnostic performance for creating OA risk assessment models in previous studies and consisted of four layers including an input layer,

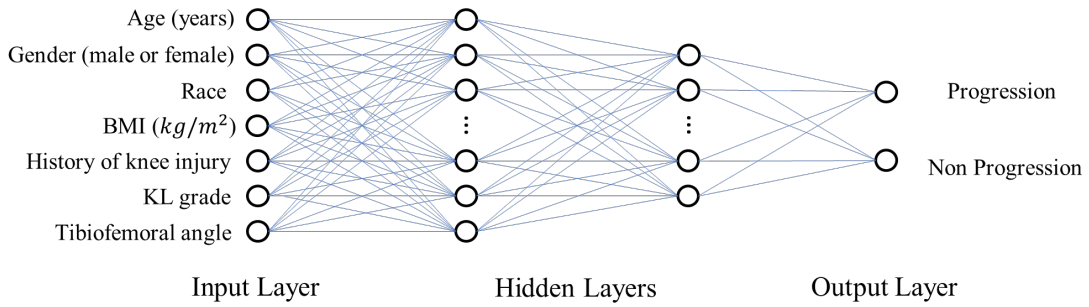

Figure 4.1: Illustration of the architecture of the traditional ANN risk assessment model for predicting the progression of radiographic joint space loss. The ANN had four layers including an input layer with seven demographic and radiographic risk factors, two hidden layers with 64 and 32 hidden nodes, and an output layer with two nodes providing a confidence value between 0 and 1 indicating the likelihood for progression of radiographic joint space loss.

two hidden layers with 64 and 32 hidden nodes, and an output layer [Amato et al. (2013); Hafezi-Nejad et al. (2017)]. The second ANN model (ANN 2) consisted of four layers including an input layer, two hidden layers with 85 and 25 hidden nodes, and an output layer [Doi (2005)]. The third ANN model (ANN 3) consisted of five layers including an input layer, three hidden layers with 20, 26 and 18 hidden nodes, and an output layer [Amato et al. (2013)]. For all ANN models, the softmax output layer was used to compress the information [Amato et al. (2013)] and provide a confidence value between 0 and 1 indicating the likelihood for progression of radiographic joint space loss. The input of the risk assessment models consisted of seven demographic and radiographic risk factors including baseline age, gender, race, BMI, history of knee injury, KL grade, and tibiofemoral angle [Hunter (2009); Silverwood et al. (2015); Bastick et al. (2015); Blagojevic et al. (2010)]. The risk factors that were continuous variables were normalized by means and standard deviations [Rajpurkar et al. (2017)]. No pre-processing of categorical variables was performed.

DL Risk Assessment Model

The DL risk assessment model consisted of two separate deep convolutional neural networks (CNNs) connected in a cascaded fashion to create a fully-automated processing pipeline. The first joint cropping CNN was used to crop regions of interest around each individual knee joint on the baseline bilateral standing posterior-anterior knee X-rays to narrow the range of information utilized for DL analysis. The second classification CNN evaluated the cropped images of the knee joint to determine the likelihood for progression of radiographic joint space loss. The detailed structure of the joint cropping and classification CNNs are described in Supplemental Table 1. The processing pipeline framework was implemented in a hybrid computing environment involving Python (version 2.7, Python Software Foundation, Wilmington, DE) and MATLAB (version 2018a, MathWorks, Natick, MA). The CNNs were coded using TensorFlow (version 1.08, Google, Mountain View, CA).

The first fully-automated joint cropping CNN was adapted from You Only Look Once (YOLO) [Redmon et al. (2016)], which consisted of 24 convolutional layers followed by two fully connected layers. The input of the CNN was the baseline knee X-rays in DICOM format, which were resized to 448×448 matrix size, normalized by the means and standard deviations of images in the ImageNet training dataset [Liu et al. (2019); Redmon et al. (2016)], and converted to NumPy arrays in Python. The convolutional layers and fully connected layers were used to extract image features to provide the coordinates of two square boxes that defined the regions of each individual knee joint on the X-rays. The pre-defined square boxes were doubled in area to correct for potential errors in the localization process and superimposed over the original DICOM X-ray images with full matrix size. Cropped images were then obtained containing each individual knee joint, which were downsized to 224×224 matrix size and used as the input to the classification CNN.

The second classification CNN was adapted from Densely Connected Convolutional Networks (DenseNet) [Huang et al. (2017)], which consisted of three dense blocks with each block connected by a convolutional layer and a maxpooling layer.

A global average pooling (GAP) layer followed the dense blocks. The SoftMax output layer with two nodes was used to compress the information 38 and provide a confidence value between 0 and 1 indicating the likelihood for progression of radiographic joint space loss. The GAP layer was modified using a gradient back-propagation method to calculate saliency maps matching the input image size that showed the regions of discriminative high activation on the X-ray on which the classification CNN based its interpretation [Zhou et al. (2016)].

Combined Traditional and DL Risk Assessment Models

Combined traditional and DL risk assessment models were developed using two different approaches. In the first approach, a simple logistic regression model was used to provide a final confidence value between 0 and 1 indicating the likelihood for progression of radiographic joint space loss based on the individual confidence values generated by the best traditional model and the DL model [Yuan and Ghosh (2008)]. The logistic regression model provided a final confidence value for the progression of radiographic joint space loss $(h(x))$ based on the two inputs:

$$h(x) = \frac{1}{1 + e^{(} - W \times X)} \tag{4.1}$$

where $X = [X_T, X_{DL}]$ was a vector of the confidence values of traditional model (X_T) and DL model (DL), and W was a vector of the parameters of the logistic regression model.

In the second approach, a joint training model was used to take into account the demographic and radiographic risk factors and the DL analysis of baseline knee X-rays as individual inputs. The combined model was developed using YOLO [Redmon et al. (2016)] and DenseNet [Huang et al. (2017); Guan et al. (2019)] to extract DL information as a feature vector, which was further concatenated with the information extracted from demographic and radiographic risk factor data. The joint training model contained three components: a feature extractor of DL analysis of baseline knee X-rays, a feature extractor of demographic and radiographic risk

factor data, and a fully connected network to combine the information. The feature extractor of DL analysis of baseline knee X-rays had the same architecture as the DL risk assessment model. The feature extractor of demographic and radiographic risk factor data was a two layer fully-connected network. The risk factor data was normalized by means and standard deviations and used as the input into a seven-dimensional fully connected layer. The output of the feature extractor of the DL analysis and the feature extractor of the risk factor data were combined as a new vector and then used as the input into another fully-connected network for joint model training. The CNNs and fully-connected layers were connected in a cascaded fashion to create a fully-automated processing pipeline as shown in Figure 4.2.

Model Training and Evaluation

Training and evaluation of the OA risk assessment models was performed on a desktop computer running a 64-bit Linux operating system (Ubuntu 16.04) with an Intel i7 7700k quad-core CPU with 32 GB DDR3 RAM and two Nvidia GTX 1080-Ti graphic cards with 3584 CUDA cores and 11GB GDDR5X RAM. A detailed description of the training and evaluation methods used for each model is provided in the Supplemental Material.

A total of 1950 knees of the 4447 knees from the 2300 subjects in the OAI database eligible to participate in our study were randomly selected for model training and evaluation, with the number chosen based upon limitations in computational efficiency and capacity. Knees with and without progression of radiographic joint space loss were randomly stratified into three non-overlapping datasets for training, validation, and hold-out testing. The randomization process was performed using a random data generator in TensorFlow (version 1.12, Google, Mountain View, CA). The training dataset consisted of 1400 knees (735 knees without and 665 knees with progression of radiographic joint space loss), which was used to iteratively optimize model parameters. The validation dataset consisted of 150 knees (76 knees without and 74 knees with progression of radiographic joint space loss), which was used to select the most optimal model during the training process. The hold-out

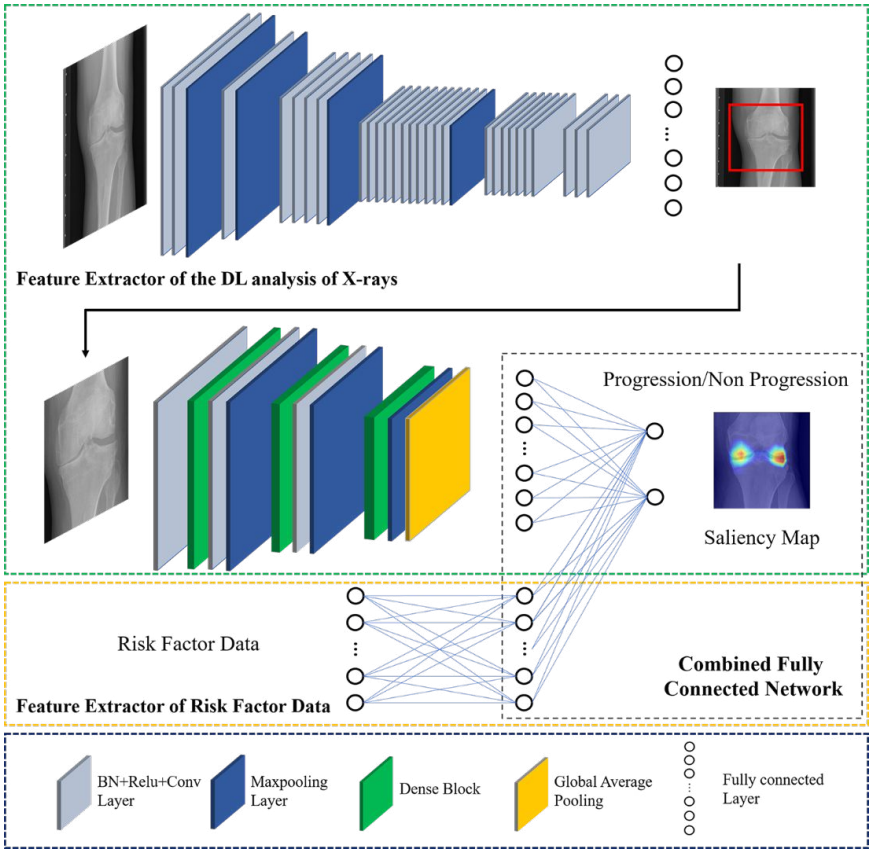

Figure 4.2: Illustration of the architecture of the combined joint training model for predicting the progression of radiographic joint space loss. The proposed model consisted of two separate convolutional neural networks connected in a cascaded fashion to create a fully-automated pipeline. The combined joint training model was created using YOLO and DenseNet to extract DL information from baseline knee X-rays as a feature vector, which was further concatenated with the normalized demographic and radiographic risk factor data vector. BN: batch normalization, Conv2D: 2D convolution, ReLU: rectified linear activation, 2D: two-dimensional.

testing dataset consisted of 400 knees (200 subjects without and 200 subjects with progression of radiographic joint space loss), which was used for final evaluation of the optimal model to avoid training over-fitting and to ensure that learned features could be generalized to new datasets. Since the outcome measure for the OA risk assessment models was the progression of medial joint space loss, the baseline X-rays of all knees with a KL grade of 3 were reviewed by a fellowship-trained musculoskeletal radiologist with 17 years of clinical experience to ensure that only knees with joint space loss more advanced in the medial than lateral compartment were included. The distribution of demographic and radiographic risk factors for knees in the training, validation, and hold-out testing datasets is provided in the Supplemental Material.

Statistical Analysis

Statistical analysis was performed using MATLAB (version 2019a, MathWorks, Natick, MA) and MedCalc (version 14.8; MedCalc Software, Ostend, Belgium). All analyzed data consisted of statistically independent observations. Statistical significance was defined as a p-value less than 0.05.

Receiver operator characteristic (ROC) analysis with areas under the curves (AUCs) was used to determine the diagnostic performance of the traditional risk assessment models, DL model, combined logistic regression model, and combined joint training model for predicting the progression of radiographic joint space loss for knees in the hold-out testing dataset. Two-sided exact binomial tests were used to calculate 95% confidence intervals. Sensitivity and specificity were also determined for the best traditional model, DL model, combined logistic regression model, and combined joint training model [Liu et al. (2019); Zhou et al. (2016)]. The Youden index was used to determine optimal model sensitivity and specificity. AUCs of the best traditional risk assessment model, DL model, combined logistic regression model, and combined joint training model were compared using a nonparametric approach [Liu et al. (2017)].

Table 4.1: Distribution of baseline KL grades for all knees, knees without progression of radiographic joint space loss, and knees with progression of radiographic joint space loss in the training dataset and hold-out testing dataset.

Knees in Training Dataset			
KL Grade	All Knees (N=1400)	Knees Without Progression (N=735)	Knees With Progression (N=665)
0	581	435	146
1	218	115	103
2	338	95	243
3	263	90	173
Knees in Validation Dataset			
KL Grade	All Knees (N=150)	Knees Without Progression (N=76)	Knees With Progression (N=74)
0	63	45	18
1	19	10	9
2	38	10	28
3	30	11	19
Knees in Hold-Out Testing Dataset			
KL Grade	All Knees (N=150)	Knees Without Progression (N=200)	Knees With Progression (N=200)
0	170	126	44
1	46	24	22
2	102	26	76
3	82	24	58

4.3 Results

Table 4.1 compares the distribution of baseline KL grades in knees without and with progression of radiographic joint space loss in both the training and testing datasets. The distribution of baseline KL grades for knees without and with progression of radiographic joint space loss was similar for the training and testing datasets. For both datasets, there were more knees without progression of radiographic joint space loss that had a baseline KL grade of 0 and more knees with progression of radiographic joint space loss that had baseline KL grades of 2 and 3.

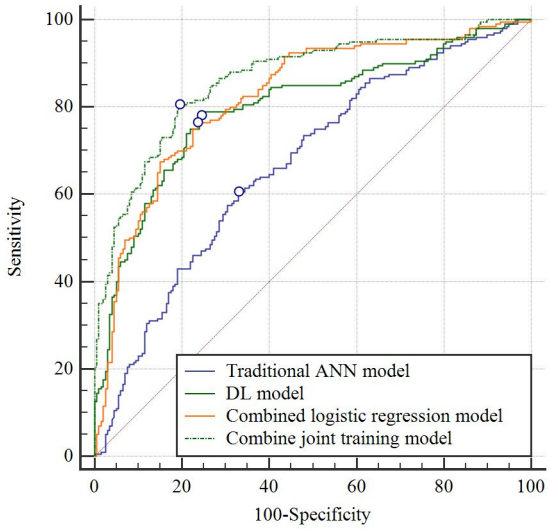

Figure 4.3: Receiver operating characteristic (ROC) curves showing the diagnostic performance of the OA risk assessment models for predicting the progression of radiographic joint space loss for knees with all baseline KL grades in the hold-out testing dataset.

The ANN model was the traditional OA risk assessment model that showed the highest diagnostic performance. The AUCs for predicting the progression of radiographic joint space loss for all knees were 0.660 (95% confidence interval of 0.620 and 0.714) for the ANN model, 0.590 (95% confidence interval of 0.540 and 0.639) for the Random forest model, and 0.572 (95% confidence interval of 0.522 and 0.621) for the logistic regression model.

Table 4.2 shows the sensitivity, specificity, and AUCs for the OA risk assessment models for predicting the progression of radiographic joint space loss in knees in the hold-out testing dataset with the ROC curves shown in Figures 4.3 and 4.4. The traditional ANN models had the lowest diagnostic performance with an AUC of 0.660 (61.5% sensitivity and 64.0% specificity) for all knees, 0.639 (64.9% sensitivity and 68.0% specificity) for KL grade 0 and 1 knees, and 0.681 (64.9% sensitivity and 68.0% specificity) for KL grades 2 and 3 knees. The combined joint training model

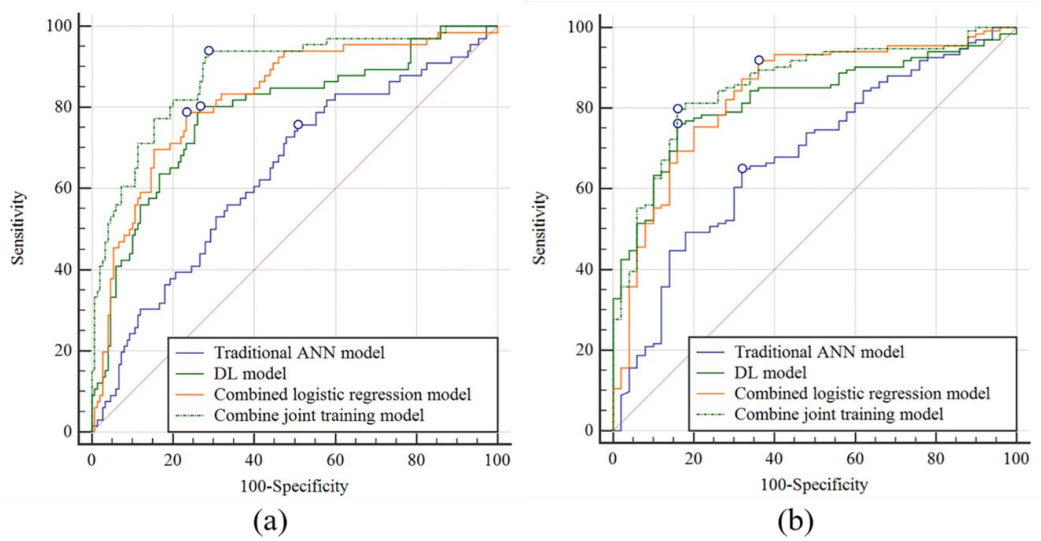

Figure 4.4: Receiver operating characteristic (ROC) curves showing the diagnostic performance of the OA risk assessment models for predicting the progression of radiographic joint space loss for knees in the hold-out testing dataset (a) without radiographic OA (baseline KL grades of 0 and 1) and (b) with radiographic OA (baseline KL grades of 2 and 3).

had the highest diagnostic performance with an AUC of 0.863 (80.5% sensitivity and specificity) for all knees, 0.882 (90.4% sensitivity and 71.3% specificity) for KL grades 0 and 1 knees, and 0.857 (79.9% sensitivity and 84.1% specificity) for KL grades 2 and 3 knees. Figure 4.5 shows saliency maps for baseline knee X-rays without and with progression of radiographic joint loss evaluated by the combined joint training model. The discriminative high activation regions on the X-rays on which the classification CNN based its interpretation were centered on the joint space and surrounding bone.

DL analysis of baseline knee X-rays improved the diagnostic performance for predicting the progression of radiographic joint space loss when compared to traditional ANN models using demographic and radiographic risk factors. The DL models, combined logistic regression models, and combined joint training models had significantly higher AUCs than the traditional ANN models for all

Table 4.2: Sensitivity, specificity, and AUCs for the OA risk assessment models for predicting the progression of radiographic joint space loss in knees in the hold-out testing dataset.

All Knees with Baseline KL Grades 0,1, 2, and 3			
(50.0% of Knees with Progression of Radiographic Joint Space Loss)			
Models	Sensitivity (95% CI)	Specificity (95% CI)	AUC (95% CI)
Traditional ANN Model	61.50% (54.4% - 68.3%)	64.00% (56.9% - 70.6%)	0.66 (0.611 - 0.706)
DL Model	78.00% (71.6% - 83.5%)	75.50% (68.9% - 81.3%)	0.799 (0.756 to 0.837)
Combined Logistic Regression Model	76.50% (70.0% - 82.2%)	76.50% (70.0% - 82.2%)	0.823 (0.781 to 0.859)
Combined Joint Training Model	80.50% (74.3% - 85.8%)	80.50% (74.3% - 85.8%)	0.863 (0.825 to 0.895)
Knees with Baseline KL Grades 0 and 1			
(30.6% of Knees with Progression of Radiographic Joint Space Loss)			
Models	Sensitivity	Specificity	AUC
Traditional ANN Model	64.90% (56.2 - 73.0)	68.00% (53.3% - 80.5%)	0.639 (0.572 to 0.704)
DL Model	80.30% (68.7% - 89.1%)	73.30% (65.5% - 80.2%)	0.787 (0.726 to 0.840)
Combined Logistic Regression Model	78.80% (67.0% - 87.9%)	70.70% (62.7% - 77.8%)	0.824 (0.767 to 0.873)
Combined Joint Training Model	90.40% (83.2% - 97.5%)	71.30% (63.4% - 78.4%)	0.882 (0.831 to 0.922)
Knees with Baseline KL Grades of 2 and 3			
(72.8% of Knees with Progression of Radiographic Joint Space Loss)			
Models	Sensitivity	Specificity	AUC
Traditional ANN Model	64.90% (56.2% - 73.0%)	68.00% (53.3% - 80.5%)	0.681 (0.608 to 0.748)
DL Model	76.10% (68.0% - 83.1%)	84.00% (70.9% - 92.8%)	0.822 (0.759 to 0.875)
Combined Logistic Regression Model	91.00% (84.9% - 95.3%)	64.10% (49.2% - 77.1%)	0.833 (0.771 to 0.884)
Combined Joint Training Model	79.90% (72.1% - 86.3%)	84.10% (70.9% - 92.8%)	0.857 (0.798 to 0.904)

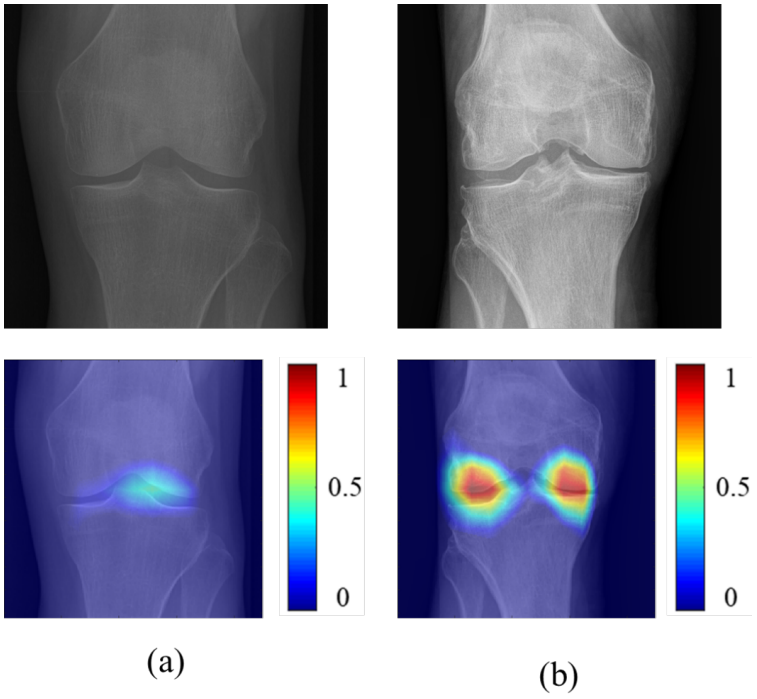

Figure 4.5: Saliency maps for baseline knee X-rays in the hold-out testing group (a) without progression of radiographic joint space loss and (b) with progression of radiographic joint loss evaluated by the combined joint training model. Note that the discriminative high activation regions on the X-rays on which the classification CNN based its interpretation were centered on the joint space and surrounding bone (color regions).

knees ($p<0.001$), KL grades 0 and 1 knees ($p<0.001$), and KL grades 2 and 3 knees ($p=0.010$-0.001). The combined joint training models had significantly higher AUCs than the DL models for all knees ($p=0.015$) and KL grades 0 and 1 knees ($p=0.006$) but not for KL grades 2 and 3 knees ($p=0.170$). There were no significant differences in AUCs between the combined logistic regression models and the DL models ($p=0.415$-0.469) or between the combined logistic regression models and combined joint training models ($p=0.183$-0.531).

4.4 Discussion

In our study, DL models were found to have significantly higher ($p<0.001$) diagnostic performance for predicting the progression of radiographic joint space loss when compared to traditional models using demographic and radiographic risk factors. The combined joint training model had the highest overall diagnostic performance with an AUC of 0.863 for predicting the progression of radiographic joint space loss for all knees. The diagnostic performance of the combined joint training model compares favorably to other risk assessment models for knee OA reported in the literature, which have had AUCs ranging between 0.70 to 0.82 for predicting OA incidence [Yoo et al. (2016); Lazzarini et al. (2017); Kerkhof et al. (2014); Zhang et al. (2011); Janvier et al. (2017); Joseph et al. (2018); DeLong et al. (1988)] and between 0.71 and 0.79 for predicting OA progression [Lazzarini et al. (2017); LaValley et al. (2017); Woloszynski et al. (2012); Janvier et al. (2017); Kraus et al. (2009)].

The combined joint training model had an AUC of 0.857 for predicting the progression of radiographic joint space loss for KL grade 2 and 3 knees in our study, which was significantly higher ($p<0.001$) than the AUC of 0.681 for the traditional model. Previous studies have also shown the benefits of analyzing baseline imaging studies in risk assessment models for predicting the progression of knee OA. LaValley et al showed that incorporating quantitative tibial subchondral bone mineral density measures on baseline dual-energy X-ray absorptiometry (DEXA) could significantly increase ($p<0.05$) the AUC of a traditional OA risk assessment model from 0.65 to 0.73 [LaValley et al. (2017)]. Studies by Janvier et al 21 and Kraus et al 22 found that incorporating quantitative subchondral tibial bone texture measures on baseline knee X-rays could significantly increase ($p<0.05$) the AUC of traditional OA risk assessment model. In these studies, the AUCs for the traditional models ranged between 0.57 and 0.71, while the AUCs of the models combining demographic and radiographic risk factors with subchondral tibial bone texture measures ranged between 0.77 and 0.79.

The combined joint training model had an AUC of 0.882 for predicting the progression of radiographic joint space loss for KL grade 0 and 1 knees in our study,

which was significantly higher ($p<0.001$) than the AUC of 0.639 for the traditional model. Previous studies have also shown the benefits of analyzing baseline imaging studies in risk assessment models for predicting the incidence of knee OA. Janvier et al showed that incorporating quantitative subchondral tibial bone texture measures on baseline knee X-rays significantly increased ($p<0.05$) the AUC of a traditional OA risk assessment model from 0.57 to 0.73 [Janvier et al. (2017)]. Joseph et al found that incorporating semi-quantitative measures of meniscal tear and cartilage lesions and quantitative measures of cartilage T2 relaxation time on baseline MR images significantly increased ($p<0.05$) the AUC of a traditional OA risk assessment models from 0.67 to 0.73 [Joseph et al. (2018)]. Unlike these previous studies, our study did not define the incidence of knee OA as the development of definitive osteophytes on knee X-rays. However, a study by Ratzlaff et al showed that the mean decrease in minimum medial joint space width in knees that transitioned from a KL grade of 0 or 1 at baseline to a KL grade of 2 at follow-up ranged between 0.18mm and 0.28mm [Riddle et al. (2016)]. Thus, it is highly likely that most KL 0 and 1 knees in our study, which showed a 0.7mm or greater decrease in minimum medial joint space width over time, also demonstrated the formation of definitive osteophytes on knee X-rays. Even if this was not the case, there is still a benefit of using knees without radiographic OA to investigate the importance of risk factors for OA progression to avoid collider bias [Ratzlaff et al. (2018)].

Previous studies have clearly shown the benefits of incorporating quantitative and semi-quantitative measures of knee joint pathology on baseline imaging studies in OA risk assessment models [LaValley et al. (2017); Janvier et al. (2017); Kraus et al. (2009); Janvier et al. (2017); Joseph et al. (2018)]. However, obtaining quantitative parameters typically requires segmenting joint structures, identifying specific features that warrant investigation based on a priori knowledge, and then extracting the features from the image datasets. Obtaining semi-quantitative parameters requires assessment of each individual structural feature on the imaging studies using a categorical based scoring system. Acquiring quantitative and semi-quantitative imaging parameters are time consuming and reader dependent and thus would be difficult to incorporate into widespread, cost-effective OA risk assessment mod-

els. One distinct advantage of the DL models developed in our study is that they can automatically learn a representative subset of features on baseline imaging studies associated with OA incidence and progression. The fully automated DL models could be widely applied in clinical practice and clinical drug trials to rapidly predict the progression of radiographic joint space loss using readily obtainable demographic and radiographic risk factors and baseline knee X-rays.

The architecture of the DL models provided high diagnostic performance for predicting the progression of radiographic joint space loss despite using a relatively small training dataset. The DenseNet classification CNN used in our study provides deeper connectivity than other neural networks, which allows direct propagation of information throughout different network layers and thereby reduces the number of parameters needed to create prediction models [Redmon et al. (2016)]. DenseNet also allows the creation of saliency maps that can be used to determine whether the regions of high activation on which the classification CNN based its interpretation are located in reasonable areas of the X-ray such as along the joint space or in regions of osteophyte formation. The weights of DenseNet in our study were also initialized by the pre-trained model of ImageNet [Ratzlaff et al. (2018)] to increase training efficiency. Finally, combined joint training was used to maximize the diagnostic performance of the DL models. Combined joint training allows the models to extract and analyze the demographic and radiographic risk factors and the DL analysis of baseline knee X-rays together to achieve the most optimal prediction performance. The same joint training approach could be used in future studies to further improve diagnostic performance by incorporating DL analysis of baseline MR images in risk assessment models for the incidence and progression of knee OA.

Our study has several limitations. One limitation was the absence of healthy knees from the OAI reference control cohort without pain or risk factors for OA in the training and testing datasets. . Another limitation was that the diagnostic performance of the OA risk assessment models was only evaluated using a hold-out testing dataset in the OAI database. Furthermore, the OA risk assessment models were only developed and evaluated for predicting the progression of radiographic

joint space loss in the medial compartment over a 48-month follow-up period. Another limitation of our study was the use of only 1950 knees in the OAI database for model training and evaluation due to limitations in computational efficiency and capacity. A final limitation was that the DL models could provide no mechanistic information regarding the factors responsible for the progression of radiographic joint space loss.

In conclusion, our study has demonstrated the feasibility of using DL risk assessment models for predicting the progression of radiographic joint space loss using baseline knee X-rays. DL models were found to have significantly higher ($p<0.001$) diagnostic performance for predicting the progression of radiographic joint space loss when compared to traditional models using demographic and radiographic risk factors. However, further validation of the DL risk assessment models is needed using different subject populations. Future work is also needed to develop more comprehensive risk assessment models incorporating DL analysis of baseline MR images for predicting the incidence and progression of knee OA.

5 VIDEO LOGO RETRIEVAL BASED ON LOCAL FEATURES

5.1 Introduction

Algorithms such as Chen et al. (2018); Joly and Buisson (2009); Revaud et al. (2012); Liao et al. (2017); Arandjelović and Zisserman (2012); Yue-Hei Ng et al. (2015); Tolias et al. (2016); Noh et al. (2017) that can retrieve, detect, or localize a target logo in a video stream [Revaud et al. (2012)] have applications in automatic annotation, dissemination impact evaluation, and suspicious logo detection [Joly and Buisson (2009); Zheng et al. (2017)]. Though several works [Chen et al. (2018); Iandola et al. (2015)] show good performance for logo retrieval, their accuracy in broadcast videos may not be acceptable for large scale commercial adoption [Joly and Buisson (2009); Revaud et al. (2012)]. Logos typically occupy only a small proportion of the frame, which can lead to failure of popular global signature-based methods such as SIFT+VLAD [Jégou et al. (2010)], SIFT+FV [Perronnin et al. (2010)] and CNNs [Zheng et al. (2017)]. Techniques for local descriptor matching are able to locate locally similar images, but suffer from a high false positive rate when used with highly semantic video frames [Liao et al. (2017)]. Logo targets are often updated frequently and so there may be inadequate training data for the application of logo recognition or detection algorithms. Many image retrieval studies are well adapted for image–to–image searching, but lack systematic estimation for image–to–video retrieval. However, image–to–video retrieval, especially convolutional based methods, usually cost more memory consumption. It is also more computational expensive and very hard to achieve real time retrieval. Therefore, an algorithm with high accuracy, memory efficiency, fast search speed, significant robustness, and modest data requirements is needed for wide adoption.

This paper proposes the VLR algorithm for Logo Retrieval outlined in Fig. 5.1. VLR contains three stages: segmentation, matching, and refinement. The process begins by down-sampling the video stream and segmenting it into regions that can be interpreted as corresponding to different camera angles or different scenes where objects can be assumed to move continuously. The matching stage analyzes

individual frames and compares to each of the target images using local feature descriptor matching, resulting in the matching matrix. The refinement stage then performs a cross analysis of all video images in the scene and exploits the continuity of information over time. In order to reduce memory consumption of CNN for image to video retrieval, we apply SpecNet to all convolutional based methods in our experiments.

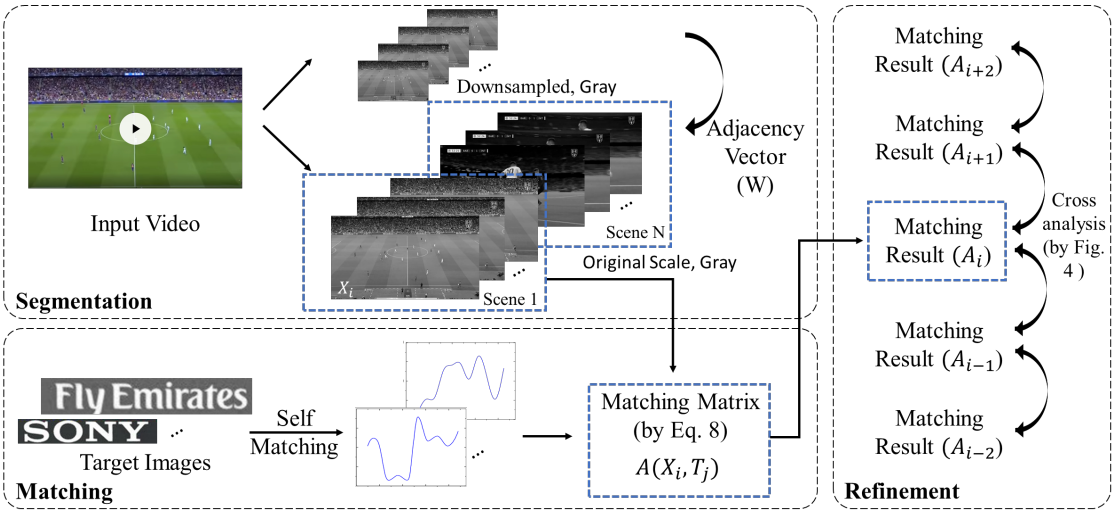

Figure 5.1: The VLR algorithm consists of three stages: video segmentation, matching, and refinement. Local features are transformed by using (5.1) to generate an adjacency vector that splits the original video into scenes. Eq. (5.3)-(5.4) are used to match target logos with videos. The refinement stage smooths the results and maintains continuity.

5.2 Related Work

Content-based Image Retrieval (CBIR) usually includes both hand-crafted descriptor-based approaches and deep learning approaches [Zheng et al. (2017)]. Before 2012, Bag-of-Words (BoW) models with hand-crafted features were predominant for image retrieval, including descriptors such as SIFT, SURF and their variants [Lowe (2004); Bay et al. (2006); Ke et al. (2004); Arandjelović and Zisserman (2012)]. These

BoW models compute statistics of the descriptors and encode images in a compact feature vector. Other BoW-like methods include VLAD [Jégou et al. (2010)] and FisherVector [Perronnin et al. (2010)], which show better performance in some applications.

With the rapid development of convolutional neural networks [Chen et al. (2018)], deep learning approaches have come to dominate [Noh et al. (2017); Liao et al. (2017)]. The performance of the deep learning methods has been improved with the appearance of strong backbone models such as Resnet [He et al. (2016)]. One promising method is DELF [Noh et al. (2017)], which designs an attention layer to extract and select the semantic local features from images, and which shows good performance compared with hand crafted descriptors. Recently, there are some interesting works focusing on logo detection in videos [Liao et al. (2017); Iandola et al. (2015)]. Unlike retrieval algorithms, these methods adopt deep features in an object detection fashion.

5.3 Video Logo Retrieval via VLR

The VLR algorithm consists of the three stages outlined in Fig. 5.1. Suppose the input is a source video with n frames, and that there are m target images of dimension (H_t, W_t). The goal is to locate these target images in the input video.

Segmentation

Videos typically contain many scenes that have little correlation. The video segmentation stage of VLR is used to segment the video into scenes that can be analyzed individually. Video segmentation contains two steps as shown in Fig. 5.1: firstly, to increase the computational efficiency, each frame is converted to grayscale and down-sampled. Let $\bar{X}_i$ be the down-sampled ith frame. Next, compute local features and match between all consecutive pairs: $\bar{X}_i$ and $\bar{X}_{i+1}$. This measure of similarity can be formalized by counting the number of detected local feature matching points divided by the total number of local feature matching keypoints.

Accordingly, let

$$p(\bar{X}_i, \bar{X}_{i+1}) = \frac{\text{\# of matching points}}{\text{total \# of keypoints}}. \tag{5.1}$$

Each term (5.1) lies between 0 and 1, and can be interpreted as a probability of two successive frames belonging to the same video scene. These are concatenated into the adjacency vector:

$$\begin{aligned} W &= \left(p(\bar{X}_1, \bar{X}_2), p(\bar{X}_2, \bar{X}_3), \ldots, p(\bar{X}_{n-1}, \bar{X}_n)\right) \\ &= (w_1, w_2, ..., w_{n-1}), \end{aligned} \tag{5.2}$$

which represents the successive similarities over time. Each element of W computes the match between neighboring frames. Fig. 5.2 shows an example, where (a) is the matching between two frames and (b) is the probability distribution between those neighboring frames. If a scene change or camera cut occurred, the number of matching keypoints between two consecutive frame will decrease. The Page-Lorden CUSUM algorithm [Hawkins and Wu (2014)] is used to detect change points in the video and to split the video frames into a collection of scenes. The following stages analyze each scene separately.

Matching

The matching stage uses SIFT or other comparable local feature sets to compute a matching matrix. Some examples of matches between targets and video images are shown in Fig. 5.3. Intuitively, targets with the most matching points will be desirable candidates. Let the number of keypoints in the target j be $N(T_j)$ and the number of matching points between T_j and the source video image i (i.e., X_i) be $N(T_j, X_i)$. As in Eq. (5.1), the empirical probability of T_j given X_i is

$$p(X_i, T_j) = \frac{N(T_j, X_i)}{N(T_j)}. \tag{5.3}$$

In Fig. 5.2(c), though $p(X_i, T_j)$ is high, it is obviously not a good matching result. Additional factors are needed.

Suppose the image sequence within a given scene has h images: $X_1, X_2, \ldots, X_h$. Partition each target image into P vertical chunks: $s = 1, ..., P$. If the target is of size $L \times K$, VLR divides the target images into P chunks of size $\frac{L \times K}{P}$. We establish a set

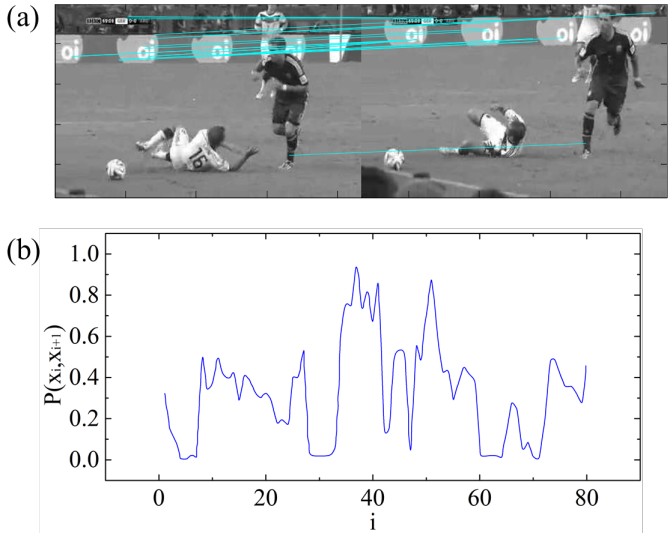

Figure 5.2: (a) shows the matching points between two neighboring images. (b) shows the matching score as a function of time. The images are divided into different segments by using Page-Lorden CUSUM algorithm [Hawkins and Wu (2014)]

of ground values by matching SIFT keypoints between the targets with themselves, which is shown in Fig. 5.2. Let $M_k(T_j)$ be the number of self-matching points in chunk k. The self-matching probability for T_j in the kth chunk is

$$p_k(T_j, T_j) = \frac{M_k(T_j)}{N(T_j, X_i)}. \tag{5.4}$$

Since each keypoint is also a matching point for the self-matching situation,

$$p(T_j, T_j) = \sum_k p_k(X_i, T_j) = 1, \tag{5.5}$$

which suggests that $p(T_j, T_j)$ can be interpreted as a matching probability distribu-

tion. This can be used as a reference when comparing the target images to other video source images.

The matching between the individual video images and the target images is again conducted using (5.4). So we can similarly use $p_k(X_i, T_j)$ to describe the keypoints matching between the target images (with P chunks) and the video images. Since not all keypoints can be matched,

$$p(x_i, T_j) = \sum_k p_k((x_i, T_j)) \leqslant 1, \tag{5.6}$$

where $p(x_i, T_j)$ is the probability of T_j with X_i. Comparing with the self-matching distribution, the correlation between these two distributions will determine if the video image contains the target. More precisely, if the correlation is weak, the target image T_j is unlikely to be in the image frame X_i even we find some matching points. The Kullback–Leibler (KL) divergence of the distributions is used to describe the correlation

$$D_{KL}(X_i, T_j) = \sum_k p_k(X_i, T_j) \log \frac{p_k(X_i, T_j)}{p_k(T_j, T_j)}. \tag{5.7}$$

Define the matching matrix $\mathbf{A}$ between the video image X_i and the target image T_j as

$$\mathbf{A}_{i,j} = [p(X_i, T_j), D_{KL}(X_i, T_j)]. \tag{5.8}$$

Then VLR uses k-$means$ clustering to pick highly distinctive matching scores from matching matrix A. If the score of both $p(X_i, T_j)$ and $D_{KL}(X_i, T_j)$ are high, the image is said to contain the target image.

Refinement

The refinement stage conducts a cross analysis of adjacent video images to determine which target images are presented. Within a single continuous scene, it is expected that targets will generally persist for a significant number of frames; it is unreasonable for a target to appear, disappear, and re-appear in a short time. Thus the raw matching result should be smoothed by this assumed continuity over time.

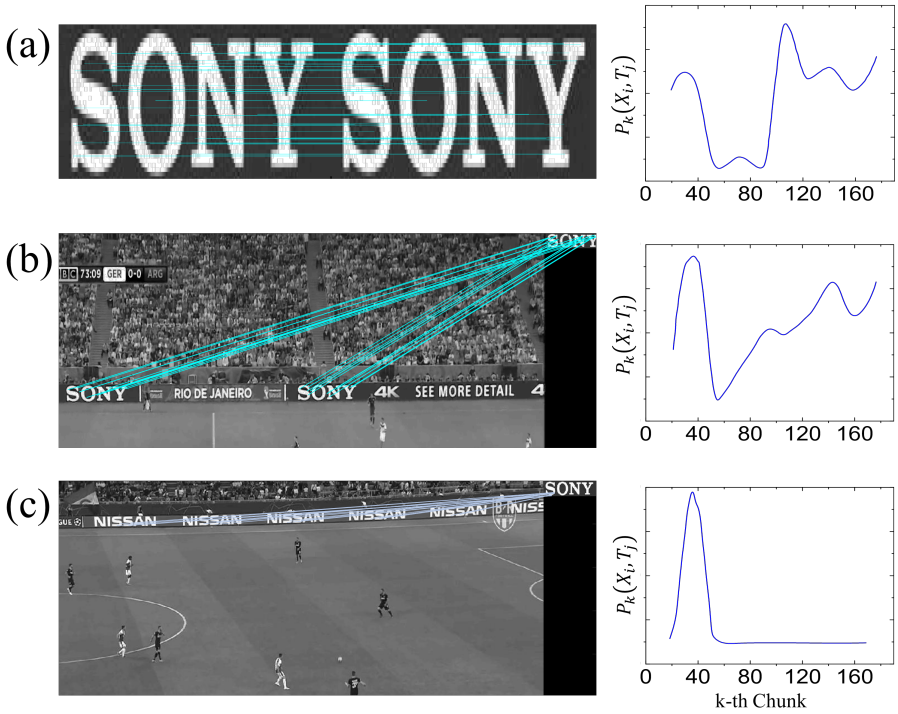

Figure 5.3: (a) shows a matching of a target image with itself. (b) shows a matching between a target image and a video frame. (c) shows an incorrect matching between a target image and a video frame despite many matching points (all the matching points are concentrated).

Assume that a sequence of video images X_i lie in a scene. We develop two parameters to smooth the estimation scores and improve the matching result: the minimum stand time t_s and the maximum lost time t_l. This cross analysis process is illustrated in Fig. 5.4.

The Minimum Standing Time t_s. If a target image appears as a candidate match for a time less than t_s, it is rejected. On the other hand, if the target image persists for a time greater than t_s, it is accepted.

The Maximum Lost Time t_l. Suppose a candidate match T_j is present but absent for time from i to $i + t$, If the interval t is less than t_l, then it is presumed that this absence is due to noise or other transient issues and it should be considered present

for the complete time.

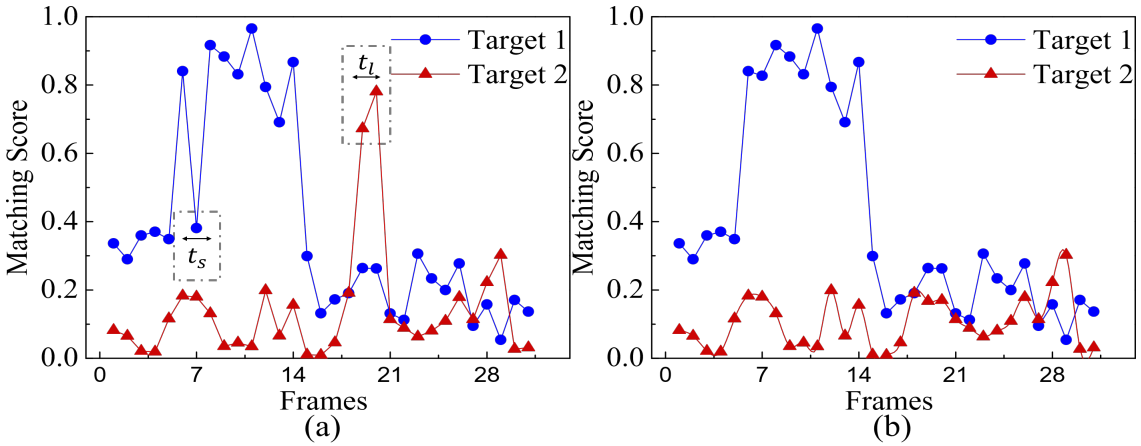

Figure 5.4: A sketch of two conditions in the refinement stage. (a) is the matching result for two different targets in a list of video frames. The time interval of targets has a sudden change. The cross analysis refines the matching result in (b).

5.4 Experiments

This section demonstrates the feasibility of VLR using two benchmark datasets. We test VLR on two different local feature extractors (SIFT, DELF) and compare the performance with several state-of-the-art algorithms: a CNN based global features algorithm (ResNet-101) [He et al. (2016)], SIFT [Lowe (2004)], DELF [Noh et al. (2017)], logo recognition algorithm [Iandola et al. (2015)], and a BoW-like algorithm (VLAD) [Jégou et al. (2010)].

Dataset and implementation details

SoccerNet is a dataset of soccer games from the European Championships with 3 seasons from 2014 to 2017. Each game consists of two untrimmed videos [Giancola et al. (2018)]. Our experiment randomly selects 45 UEFA Champions League (UEFA) games (15 from each season). We down-sample one frame a second from the source

Table 5.1: Comparison of proposed techniques against state-of-the-art algorithms on SoccerNet dataset in three different seasons.

Model	SoccerNet (mAP)		
	2014-2015	2015-2016	2016-2017
ResNet101-GeM [He et al. (2016)]	67.5	65.2	73.6
SIFT [Lowe (2004)]	69.5	65.3	75.2
VLAD [Jégou et al. (2010)]	70.6	66.4	72.8
DELF [Noh et al. (2017)]	74.4	71.3	78.1
Deeplogo [Iandola et al. (2015)]	78.9	72.3	73.8
Logo Proposals [Qi et al. (2017)]	69.8	68.2	73.7
One-Shot Network [Bhunia et al. (2019)]	71.2	70.4	71.6
VLR (SIFT)	73.2	79.8	78.9
VLR (DELF)	80.6	80.7	83.5

video images, testing a total of 248,400 images with resolution of 1920×1080 pixels. Since the sponsors of UEFA vary for different years, we use seven advertisements sponsored each year from 2014 to 2017 as the targets.

Stanford I2V is a video database consisting of newscast videos annotated with more than 200 ground-truth queries [Araujo et al. (2015)]. Our experiment selects 200 news videos for testing, most last no longer than 2 minutes. All the videos are down-sampled to one frame per half second.

VLR was implemented using SIFT features in Matlab 2016a. For VLR of DELF, features are first computed using Python/Tensorflow, and the matching is computed by using the same technique as in Section 5.3. The comparison of experiments of VLAD and SIFT are conducted in Matlab, while the deep learning algorithms are run in Python/Tensorflow. Target logos are usually updated every season/year. Since there are only a limited images of new logos, they cannot provide an adequate dataset for training alone. Instead, the deep learning models are trained using the OpenLogo dataset [Agarwal et al. (2018)], which includes many of the target logo images needed in the experiments.

Table 5.2: Comparison of proposed techniques against state-of-the-art algorithms on Stanford I2V dataset in two logo retrieval tasks.

Model	Stanford I2V (mAP)	
	TV Channel Logo	Social Software Logo
ResNet101 [He et al. (2016)]	79.5	30.3
SIFT [Lowe (2004)]	83.4	38.6
VLAD [Jégou et al. (2010)]	82.6	41.7
DELF [Noh et al. (2017)]	84.5	42.3
Deeplogo [Iandola et al. (2015)]	87.1	43.1
Logo Proposals [Qi et al. (2017)]	84.9	42.5
One-Shot Network [Bhunia et al. (2019)]	83.5	40.7
VLR (SIFT)	87.4	41.7
VLR (DELF)	89.2	46.8

Results

Table 5.1 shows the mean Average Precision (mAP) of retrieval for each logo target in the UEFA videos, compared to a ground truth which was determined by manually counting the time that each target was visible. VLR based on DELF achieves more than 80% mAP over the three year's dataset, which is significantly better than the other models. Both VLR based on SIFT and DELF improve at least 5% over the original SIFT and DELF algorithms. Since many regions of the video are irrelevant, models with local features benefit from the small scaled targets. While the logo recognition algorithms show good performance in some cases, they tend to be less consistent and less reliable, likely being influenced by the correlation between the pre-trained dataset and the experimental test dataset.

Table 5.2 shows experimental results on the Stanford I2V dataset. The test dataset contains 200 broadcast news videos. Two tasks are tested: 1) Retrieval of three TV channel logos (CNN, CBS and Bloomberg) and 2) Retrieval of social media logos (such as Twitter and Facebook). The VLR based on DELF achieves the highest mAP for both tasks. VLR based on SIFT or DELF can achieve improvement of 3% to 5% over the original SIFT and DELF algorithms.

5.5 Conclusion

This paper introduced a logo image retrieval algorithm based on local features, which consists of three stages: segmentation, matching, and refinement. The algorithm was tested and verified on two benchmark datasets and shows notable performance advantages compared with several state of the art methods. The idea of the VLR is not limited to SIFT or DELF, and it has potential to improve performance using other local features.

6 FUTURE WORK

We proposed a new convolutional network architecture, which we refer to as Spectral Domain Convolutional Neural Network (SpecNet). It optimizes memory usage during training and testing by converting feature maps into the frequency domain while leaving the convolutional kernels in the spatial domain during both forward and backward propagation. The kernels are converted whenever needed into the frequency domain, and the memory required can then be released immediately after use. In addition, we designed the activation function for SpecNet. We show that this approach can reduce memory usage by about 63% without adversely affecting the performance when compared with with other memory-efficient CNN algorithms. There are a number of ideas we would like to investigate to improve SpecNet even further.

Network optimization

Layer Normalization. It is common to reduce training epochs by including an extra normalization layer in deep neural networks. The normalization standardizes each summed input using its mean and standard deviation across the input data [Ba et al. (2016); Rajpurkar et al. (2017); Liu et al. (2019)]. Deep learning neural networks trained using batch normalization often converge faster even with simple SGD. In addition to training time improvement, the stochasticity inherited from the batch statistics can serve as a regularizer during training.

Feature maps in the spectrum domain are very different from those in the spatial domain. It is crucial to maintain symmetry of feature maps in spectrum domain. Besides, we find it hard to design the strict equivalent batch normalization in spectrum domain and layer normalization has a strong influence on training. Therefore, in our experiments, we did not apply batch normalization to SpecNet and also remove batch normalization layers for comparison CNNs. It's worthwhile to design a specific layer normalization method for SpecNet, since the convergence speed of SpecNet is slightly slower than CNNs in spatial domain.

Activation function. For SpecNet, activation functions must fulfill the following design guidelines:

1. They allow inexpensive gradient calculation.
2. Both $g(x)$ and $h(x)$ are monotonic nondecreasing
3. The functions are odd, i.e. $g(-x) = -g(x)$.

Based on these, we may design several activation functions that may be applied in SpecNet. For example, a piecewise function may work for this problem:

$$f(x) = \begin{cases} x, & x > \alpha \\ 0, & -\alpha < x < \alpha \\ -x, & x < -\alpha \end{cases} \tag{6.1}$$

for $\alpha \in \mathbb{R}^+$, or more generally,

$$f_\beta(x) = \begin{cases} x^\beta, & x > \alpha \\ 0, & -\alpha < x < \alpha \\ -x^\beta, & x < -\alpha \end{cases} \tag{6.2}$$

for $\beta > 0$. For noninteger β, $f_\beta(x)$ would assume the positive real root.

Thus finding functions that satisfy the three criteria is not difficult. However, it is not easy to know *a priori* whether a given activation function will provide for efficient training. Poor activation functions may have vanishing gradients, may saturate, or may result in an excessive number of local minimal. Experiments suggest that values influenced by the activation function in the spatial and spectrum domains are not the same. Figure 6.1 shows the output of a converged VGG16 feature map before and after the activation function. Note that half of the values in the spatial domain are affected by the activation function (Relu) in spatial domain and about a forth of values are affected by our activation function in SpecNet. Thus an optimal activation function must consider the overall magnitude of the signals (and hence the normalization strategy that is used) as well as ensuring that

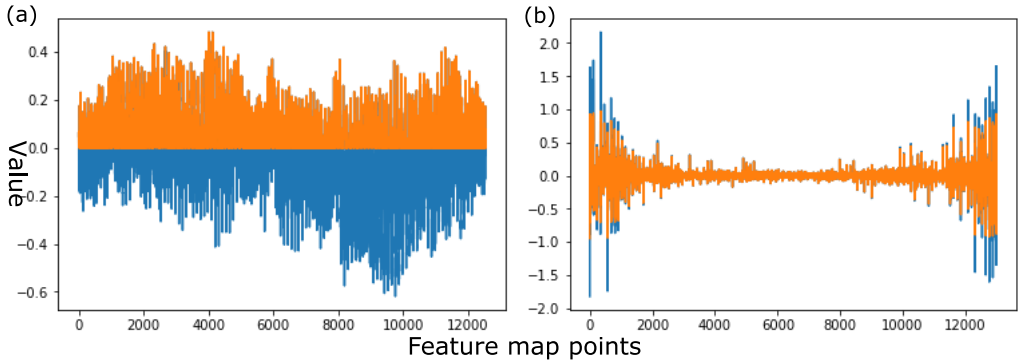

Figure 6.1: (a) Values of a VGG16 feature map in spatial domain. (b) Values of a feature map in SpecVGG16. The blue/red points are feature map values before/after the activation function.

a significant percentage of the values lie outside the (relatively) linear region of the activation function. Determining that an activation is optimal would require a significant number of further experiments.

Application

From Fig. 2.2, SpecNet can achieve significant testing accuracy, fast computation speed and memory efficiency during training, which makes it a good candidate to implement deeper CNNs and 3D CNNs. We had shown two applications based on SpecNet in Chapters 3 and 4. These rely on SpecNet to significantly reduce memory consumption and enable the CNNs to solve new problems even with limited memory. In addition, there are other applications that may benefit from the SpecNet idea.

Other deep learning algorithms based on CNNs. We have already applied SpecNet to several image classification CNNs such as AlexNet, VGG-16, DenseNet and et. These algorithms have multiple convolution computations and show the performance of SpecNet directly. Several object detection deep learning algorithms such as YOLO, FastRCNN, also based on CNN to extract features from input images. Image segmentation algorithms such as U-Net using CNN to encode and decode

to generate segmented mask. In principle, all algorithms based on CNNs can use SpecNet to reduce its memory consumption. To help others make use of the SpecNet idea, we can envision a CUDA API for Tensorflow. This could help anyone to optimize the memory usage of their convolutional layers.

3D convolution algorithms. We have already shown two medical applications of SpecNet using MRI and X-ray images that show very good performance. In addition, we demonstrated a logo video retrieval algorithm in videos (VLD) based on SpecNet. However, these applications semi-3D CNNs but do not use 3D convolutions to combine information in different channels. But semi-3D cannot fully extract features in different channels and these feature may be important in some applications. For example, VLD and existing 2D convolutional neural networks are blind to much of this kind of information in the video sequence. 3D convolution may be one approach, but its memory cost is huge, and large memory usage places an effective limit on the amount of training or testing. SpecNet enables a possible solution by combining 3D convolution layers with 2D convolution as a new network structure.

DCT and DWT

Our work has shown that converting feature maps into the spectral domain can compress memory usage without reducing accuracy significantly. Therefore, similar operations such as discrete cosine transform (DCT) and Discrete wavelet transform (DWT) may also be choices for converting feature map. It should be noticed that feature maps in the spectral domain are easily applied to convolution and some papers [Pratt et al. (2017); Mathieu et al. (2013); Fujieda et al. (2017)] show how the FFT can be applied to accelerate computation. Feature maps after DCTs and DWTs may be still real numbers (saving memory for imaginary part) and even more sparse. But the convolution computation is more complex. Therefore, it is worth investigating if converting to DCT or DWT will decrease the memory (or not) and if such conversion is sensible.

REFERENCES

Agarwal, Shivang, Jean Ogier Du Terrail, and Frédéric Jurie. 2018. Recent advances in object detection in the age of deep convolutional neural networks. *arXiv preprint arXiv:1809.03193*.

Amato, Filippo, Alberto López, Eladia María Peña-Méndez, Petr Vaňhara, Aleš Hampl, and Josef Havel. 2013. Artificial neural networks in medical diagnosis.

Andri, R., L. Cavigelli, D. Rossi, and L. Benini. 2018. Yodann: An architecture for ultralow power binary-weight cnn acceleration. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 37(1):48–60.

Anthimopoulos, Marios, Stergios Christodoulidis, Lukas Ebner, Andreas Christe, and Stavroula Mougiakakou. 2016. Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network. *IEEE Transactions on Medical Imaging* 35(5):1207–1216.

Arandjelović, Relja, and Andrew Zisserman. 2012. Three things everyone should know to improve object retrieval. In *2012 ieee conference on computer vision and pattern recognition*, 2911–2918. IEEE.

Araujo, André, Jason Chaves, David Chen, Roland Angst, and Bernd Girod. 2015. Stanford i2v: A news video dataset for query-by-image experiments. In *Proceedings of the 6th acm multimedia systems conference*, 237–242. MMSys '15, ACM.

Arevalo, John, Fabio A. González, Raúl Ramos-Pollán, Jose L. Oliveira, and Miguel Angel Guevara Lopez. 2016. Representation learning for mammography mass lesion classification with convolutional neural networks. *Computer Methods and Programs in Biomedicine* 127:248–257.

Ayat, Sayed Omid, Mohamed Khalil-Hani, Ab Al-Hadi Ab Rahman, and Hamdan Abdellatef. 2019. Spectral-based convolutional neural network without multiple spatial-frequency domain switchings. *Neurocomputing* 364:152–167.

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Bastick, Alex N, Janneke N Belo, Jos Runhaar, and Sita MA Bierma-Zeinstra. 2015. What are the prognostic factors for radiographic progression of knee osteoarthritis? a meta-analysis. *Clinical Orthopaedics and Related Research®* 473(9):2969–2989.

Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In *European conference on computer vision*, 404–417. Springer.

Bedson, John, Kelvin Jordan, and Peter Croft. 2005. The prevalence and history of knee osteoarthritis in general practice: a case–control study. *Family practice* 22(1): 103–108.

Bhunia, Ayan Kumar, Ankan Kumar Bhunia, Shuvozit Ghose, Abhirup Das, Partha Pratim Roy, and Umapada Pal. 2019. A deep one-shot network for query-based logo retrieval. *Pattern Recognition* 96:106965.

Bien, Nicholas, Pranav Rajpurkar, Robyn L. Ball, Jeremy Irvin, Allison Park, et al. 2018. Deep-learning-assisted diagnosis for knee magnetic resonance imaging: Development and retrospective validation of MRNet. *PLOS Medicine* 15(11):e1002699.

Blagojevic, M, C Jinks, A Jeffery, and 1KP Jordan. 2010. Risk factors for onset of osteoarthritis of the knee in older adults: a systematic review and meta-analysis. *Osteoarthritis and cartilage* 18(1):24–33.

Bottou, Léon. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of compstat'2010*, 177–186. Springer.

Changpinyo, Soravit, Mark Sandler, and Andrey Zhmoginov. 2017. The power of sparsity in convolutional neural networks. *CoRR* abs/1702.06257.

Chao, Ping, Chao-Yang Kao, Yu-Shan Ruan, Chien-Hsiang Huang, and Youn-Long Lin. 2019. Hardnet: A low memory traffic network. In *The ieee international conference on computer vision (iccv)*.

Chen, Tianqi, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. 2016. Training deep nets with sublinear memory cost. *CoRR* abs/1604.06174.

Chen, Ziqian, Jie Lin, Vijay Chandrasekhar, and Ling-Yu Duan. 2018. Gated square-root pooling for image instance retrieval. In *2018 25th ieee international conference on image processing (icip)*, 1982–1986. IEEE.

Cheng, Yu, Duo Wang, Pan Zhou, and Tao Zhang. 2017. A survey of model compression and acceleration for deep neural networks. *CoRR* abs/1710.09282.

Cicuttini, Flavia Maria, Anita E Wluka, Y Wang, and SL Stuckey. 2004. Longitudinal study of changes in tibial and femoral cartilage in knee osteoarthritis. *Arthritis & Rheumatism: Official Journal of the American College of Rheumatology* 50(1):94–97.

Ciompi, Francesco, Bartjan de Hoop, Sarah J. van Riel, Kaman Chung, Ernst Th. Scholten, Matthijs Oudkerk, Pim A. de Jong, Mathias Prokop, and Bram van Ginneken. 2015. Automatic classification of pulmonary peri-fissural nodules in computed tomography using an ensemble of 2D views and a convolutional neural network out-of-the-box. *Medical Image Analysis* 26(1):195–202.

Courbariaux, Matthieu, and Yoshua Bengio. 2016. Binarynet: Training deep neural networks with weights and activations constrained to +1 or -1. *CoRR* abs/1602.02830.

De Fauw, Jeffrey, Joseph R Ledsam, Bernardino Romera-Paredes, Stanislav Nikolov, et al. 2018. Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature medicine* 24(9):1342–1350.

DeLong, E R, D M DeLong, and D L Clarke-Pearson. 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics* 44(3):837–45.

Denton, Emily L, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus. 2014. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in neural information processing systems 27*, ed. Z. Ghahramani,

M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, 1269–1277. Curran Associates, Inc.

Doi, Kunio. 2005. Current status and future potential of computer-aided diagnosis in medical imaging. *The British journal of radiology* 78(suppl_1):s3–s19.

Felson, David T. 2004. An update on the pathogenesis and epidemiology of osteoarthritis. *Radiologic Clinics* 42(1):1–9.

Felson, David T, T Derek V Cooke, Jingbo Niu, Joyce Goggins, John Choi, Joseph Yu, Michael C Nevitt, and The OAI Investigators Group. 2009. Can anatomic alignment measured from a knee radiograph substitute for mechanical alignment from full limb films? *Osteoarthritis and cartilage* 17(11):1448–1452.

Felson, David T, and Richard Hodgson. 2014. Identifying and treating preclinical and early osteoarthritis. *Rheumatic Disease Clinics* 40(4):699–710.

Felson, David T, and Yuqing Zhang. 1998. An update on the epidemiology of knee and hip osteoarthritis with a view to prevention. *Arthritis & Rheumatism: Official Journal of the American College of Rheumatology* 41(8):1343–1355.

Fluss, Ronen, David Faraggi, and Benjamin Reiser. 2005. Estimation of the Youden Index and its Associated Cutoff Point. *Biometrical Journal* 47(4):458–472.

Fujieda, Shin, Kohei Takayama, and Toshiya Hachisuka. 2017. Wavelet convolutional neural networks for texture classification. *CoRR* abs/1707.07394.

Gandy, SJ, PA Dieppe, MC Keen, RA Maciewicz, I Watt, and JC Waterton. 2002. No loss of cartilage volume over three years in patients with knee osteoarthritis as assessed by magnetic resonance imaging. *Osteoarthritis and cartilage* 10(12): 929–937.

Giancola, Silvio, Mohieddine Amine, Tarek Dghaily, and Bernard Ghanem. 2018. Soccernet: A scalable dataset for action spotting in soccer videos. In *Proceedings of the ieee conference on computer vision and pattern recognition workshops*, 1711–1721.

Gianotti, Simon M., Stephen W. Marshall, Patria A. Hume, and Lorna Bunt. 2009. Incidence of anterior cruciate ligament injury and other knee ligament injuries: A national population-based study. *Journal of Science and Medicine in Sport* 12(6): 622–627.

Guan, Bochen, Jinnian Zhang, William A Sethares, Richard Kijowski, and Fang Liu. 2019. Specnet: Spectral domain convolutional neural network. *arXiv preprint arXiv:1905.10915*.

Gupta, Suyog, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. 2015. Deep learning with limited numerical precision. In *International conference on machine learning*, 1737–1746.

Hafezi-Nejad, Nima, Ali Guermazi, Frank W Roemer, David J Hunter, Erik B Dam, Bashir Zikria, C Kent Kwoh, and Shadpour Demehri. 2017. Prediction of medial tibiofemoral compartment joint space loss progression using volumetric cartilage measurements: data from the fnih oa biomarkers consortium. *European radiology* 27(2):464–473.

Halilaj, Eni, Ya Le, JL Hicks, TJ Hastie, and SL Delp. 2018. Modeling and predicting osteoarthritis progression: data from the osteoarthritis initiative. *Osteoarthritis and cartilage* 26(12):1643–1650.

Hanna, Fahad, Peter Robert Ebeling, Yuanyuan Wang, Richard OâŁ™Sullivan, Susan Davis, Anita E Wluka, and Flavia Maria Cicuttini. 2005. Factors influencing longitudinal change in knee cartilage volume measured from magnetic resonance imaging in healthy men. *Annals of the rheumatic diseases* 64(7):1038–1042.

Hanson, Stephen Jose, and Lorien Y. Pratt. 1989. Comparing biases for minimal network construction with back-propagation. In *Advances in neural information processing systems 1*, ed. D. S. Touretzky, 177–185. Morgan-Kaufmann.

Hawkins, Douglas M, and Qifan Wu. 2014. The cusum and the ewma head-to-head. *Quality Engineering* 26(2):215–222.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Hinton, Geoffrey E, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Huang, Gao, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. 2017. Densely connected convolutional networks. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Huang, Lei, Yan Jin, Yaozong Gao, Kim-Han Thung, Dinggang Shen, Alzheimer's Disease Neuroimaging Initiative, et al. 2016. Longitudinal clinical score prediction in alzheimer's disease with soft-split sparse regression based random forest. *Neurobiology of aging* 46:180–191.

Hunter, DJ. 2009. Risk stratification for knee osteoarthritis progression: a narrative review. *Osteoarthritis and cartilage* 17(11):1402–1407.

Iandola, Forrest N, Anting Shen, Peter Gao, and Kurt Keutzer. 2015. Deeplogo: Hitting logo recognition with the deep neural network hammer. *arXiv preprint arXiv:1510.02131*.

Jain, Animesh, Amar Phanishayee, Jason Mars, Lingjia Tang, and Gennady Pekhimenko. 2018. Gist: Efficient data encoding for deep neural network training. In *45th ACM/IEEE annual international symposium on computer architecture, ISCA 2018, los angeles, ca, usa, june 1-6, 2018*, 776–789.

Janvier, T, R Jennane, H Toumi, and E Lespessailles. 2017. Subchondral tibial bone texture predicts the incidence of radiographic knee osteoarthritis: data from the osteoarthritis initiative. *Osteoarthritis and cartilage* 25(12):2047–2054.

Jégou, Hervé, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. 2010. Aggregating local descriptors into a compact image representation. In *Cvpr 2010-23rd*

ieee conference on computer vision & pattern recognition, 3304–3311. IEEE Computer Society.

Joly, Alexis, and Olivier Buisson. 2009. Logo retrieval with a contrario visual query expansion. In *Proceedings of the 17th acm international conference on multimedia*, 581–584. MM '09, New York, NY, USA: ACM.

Joseph, Gabby B, Charles E McCulloch, Michael C Nevitt, Jan Neumann, Alexandra S Gersing, Martin Kretzschmar, Benedikt J Schwaiger, John A Lynch, Ursula Heilmeier, Nancy E Lane, et al. 2018. Tool for osteoarthritis risk prediction (toarp) over 8 years using baseline clinical data, x-ray, and mri: Data from the osteoarthritis initiative. *Journal of Magnetic Resonance Imaging* 47(6):1517–1526.

Karsdal, MA, M Michaelis, C Ladel, AS Siebuhr, AR Bihlet, JR Andersen, H Guehring, C Christiansen, AC Bay-Jensen, and VB Kraus. 2016. Disease-modifying treatments for osteoarthritis (dmoads) of the knee and hip: lessons learned from failures and opportunities for the future. *Osteoarthritis and Cartilage* 24(12):2013–2021.

Ke, Yan, Rahul Sukthankar, et al. 2004. Pca-sift: A more distinctive representation for local image descriptors. *CVPR (2)* 4:506–513.

Kerkhof, HJM, SMA Bierma-Zeinstra, NK Arden, Sarah Metrustry, Martha Castano-Betancourt, DJ Hart, Albert Hofman, F Rivadeneira, EHG Oei, Tim D Spector, et al. 2014. Prediction model for knee osteoarthritis incidence, including clinical, genetic and biochemical risk factors. *Annals of the rheumatic diseases* 73(12): 2116–2121.

Kingma, Diederik P., and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *ArXiv e-prints*.

Kooi, Thijs, Geert Litjens, Bram van Ginneken, Albert Gubern-Mérida, Clara I. Sánchez, Ritse Mann, Ard den Heeten, and Nico Karssemeijer. 2017. Large scale deep learning for computer aided detection of mammographic lesions. *Medical Image Analysis* 35:303–312.

Kothari, Manish, Ali Guermazi, Gabriele von Ingersleben, Yves Miaux, Martine Sieffert, Jon E Block, Randall Stevens, and Charles G Peterfy. 2004. Fixed-flexion radiography of the knee provides reproducible joint space width measurements in osteoarthritis. *European radiology* 14(9):1568–1573.

Kraus, Virginia Byers, Sheng Feng, ShengChu Wang, Scott White, Maureen Ainslie, Alan Brett, Anthony Holmes, and H Cecil Charles. 2009. Trabecular morphometry by fractal signature analysis is a novel marker of osteoarthritis progression. *Arthritis & Rheumatism: Official Journal of the American College of Rheumatology* 60(12): 3711–3722.

Krizhevsky, Alex, and Geoffrey Hinton. 2009. Learning multiple layers of features from tiny images. Tech. Rep., Citeseer.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012a. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems 25*, ed. F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, 1097–1105. Curran Associates, Inc.

———. 2012b. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems 25*, ed. F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, 1097–1105. Curran Associates, Inc.

Lakhani, Paras, and Baskaran Sundaram. 2017. Deep Learning at Chest Radiography: Automated Classification of Pulmonary Tuberculosis by Using Convolutional Neural Networks. *Radiology* 284(2):574–582.

LaValley, Michael P, Grace H Lo, Lori Lyn Price, Jeffrey B Driban, Charles B Eaton, and Timothy E McAlindon. 2017. Development of a clinical prediction algorithm for knee osteoarthritis structural progression in a cohort study: value of adding measurement of subchondral bone density. *Arthritis research & therapy* 19(1):95.

Lazzarini, N, Jos Runhaar, AC Bay-Jensen, CS Thudium, SMA Bierma-Zeinstra, Yves Henrotin, and J Bacardit. 2017. A machine learning approach for the identifi-

cation of new biomarkers for knee osteoarthritis development in overweight and obese women. *Osteoarthritis and cartilage* 25(12):2014–2021.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521(7553):436.

LeCun, Yann, LD Jackel, Leon Bottou, A Brunot, Corinna Cortes, JS Denker, Harris Drucker, I Guyon, UA Muller, Eduard Sackinger, et al. 1995. Comparison of learning algorithms for handwritten digit recognition. In *International conference on artificial neural networks*, vol. 60, 53–60. Perth, Australia.

Lee, J K, L Yao, C T Phelps, C R Wirth, J Czajka, and J Lozman. 1988. Anterior cruciate ligament tears: MR imaging compared with arthroscopy and clinical tests. *Radiology* 166(3):861–864.

Lester, Grant. 2008. Clinical research in oa–the nih osteoarthritis initiative. *J Musculoskelet Neuronal Interact* 8(4):313–314.

Li, Hao, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. 2016. Pruning filters for efficient convnets. *CoRR* abs/1608.08710.

Li, Yuchao, Shaohui Lin, Baochang Zhang, Jianzhuang Liu, David Doermann, Yongjian Wu, Feiyue Huang, and Rongrong Ji. 2019. Exploiting kernel sparsity and entropy for interpretable cnn compression. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Liao, Yuan, Xiaoqing Lu, Chengcui Zhang, Yongtao Wang, and Zhi Tang. 2017. Mutual enhancement for detection of multiple logos in sports videos. In *Proceedings of the ieee international conference on computer vision*, 4846–4855.

Liu, Baoyuan, Min Wang, Hassan Foroosh, Marshall Tappen, and Marianna Pensky. 2015. Sparse convolutional neural networks. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Liu, Fang. 2018. SUSAN: segment unannotated image structure using adversarial network. *Magnetic Resonance in Medicine*.

———. 2019. Susan: segment unannotated image structure using adversarial network. *Magnetic resonance in medicine* 81(5):3330–3345.

Liu, Fang, Bochen Guan, Zhaoye Zhou, Alexey Samsonov, Humberto Rosas, Kevin Lian, Ruchi Sharma, Andrew Kanarek, John Kim, Ali Guermazi, et al. 2019. Fully automated diagnosis of anterior cruciate ligament tears on knee mr images by using deep learning. *Radiology: Artificial Intelligence* 1(3):180091.

Liu, Fang, Zhaoye Zhou, Hyungseok Jang, Alexey Samsonov, Gengyan Zhao, and Richard Kijowski. 2017. Deep Convolutional Neural Network and 3D Deformable Approach for Tissue Segmentation in Musculoskeletal Magnetic Resonance Imaging. *Magnetic resonance in medicine* DOI: 10.1002/mrm.26841.

Liu, Fang, Zhaoye Zhou, Alexey Samsonov, Donna Blankenbaker, Will Larison, Andrew Kanarek, Kevin Lian, Shivkumar Kambhampati, and Richard Kijowski. 2018. Deep Learning Approach for Evaluating Knee MR Images: Achieving High Diagnostic Performance for Cartilage Lesion Detection. *Radiology* 172986.

Long, Jonathan, Evan Shelhamer, and Trevor Darrell. 2014. Fully Convolutional Networks for Semantic Segmentation.

Lowe, David G. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision* 60(2):91–110.

Malon, Christopher D, and Eric Cosatto. 2013. Classification of mitotic figures with convolutional neural networks and seeded blob features. *Journal of pathology informatics* 4:9.

Mathieu, Michael, Mikael Henaff, and Yann LeCun. 2013. Fast training of convolutional networks through ffts. *arXiv preprint arXiv:1312.5851*.

Meng, Chen, Minmin Sun, Jun Yang, Minghui Qiu, and Yang Gu. 2017. Training deeper models by gpu memory optimization on tensorflow. In *Proc. of ml systems workshop in nips*.

Netzer, Yuval, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. 2011. Reading digits in natural images with unsupervised feature learning. In *Nips workshop on deep learning and unsupervised feature learning 2011*.

Neumann, G, D Hunter, M Nevitt, LB Chibnik, K Kwoh, H Chen, T Harris, S Satterfield, J Duryea, et al. 2009. Location specific radiographic joint space width for osteoarthritis progression. *Osteoarthritis and cartilage* 17(6):761–765.

Noh, Hyeonwoo, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. 2017. Large-scale image retrieval with attentive deep local features. In *Proceedings of the ieee international conference on computer vision*, 3456–3465.

Norman, Berk, Valentina Pedoia, and Sharmila Majumdar. 2018. Use of 2d u-net convolutional neural networks for automated cartilage and meniscus segmentation of knee mr imaging data to determine relaxometry and morphometry. *Radiology* 288(1):177–185.

Peat, G, R McCarney, and P Croft. 2001. Knee pain and osteoarthritis in older adults: a review of community burden and current use of primary health care. *Annals of the rheumatic diseases* 60(2):91–97.

Pedoia, Valentina, Berk Norman, Sarah N. Mehany, Matthew D. Bucknor, Thomas M. Link, and Sharmila Majumdar. 2018. 3D convolutional neural networks for detection and severity staging of meniscus and PFJ cartilage morphological degenerative changes in osteoarthritis and anterior cruciate ligament subjects. *Journal of Magnetic Resonance Imaging*.

Perronnin, Florent, Jorge Sánchez, and Thomas Mensink. 2010. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, 143–156. Springer.

Pleiss, Geoff, Danlu Chen, Gao Huang, Tongcheng Li, Laurens van der Maaten, and Kilian Q Weinberger. 2017. Memory-efficient implementation of densenets. *arXiv preprint arXiv:1707.06990*.

Pratt, Harry, Bryan M. Williams, Frans Coenen, and Yalin Zheng. 2017. Fcnn: Fourier convolutional neural networks. In *Ecml/pkdd*.

Qi, C., C. Shi, C. Wang, and B. Xiao. 2017. Logo retrieval using logo proposals and adaptive weighted pooling. *IEEE Signal Processing Letters* 24(4):442–445.

Rajpurkar, Pranav, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. 2017. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*.

Ratzlaff, Charles, Erin L Ashbeck, Ali Guermazi, Frank W Roemer, Jeffrey Duryea, and Chian K Kwoh. 2018. A quantitative metric for knee osteoarthritis: reference values of joint space loss. *Osteoarthritis and cartilage* 26(9):1215–1224.

Raynauld, Jean-Pierre, Johanne Martel-Pelletier, Marie-Josée Berthiaume, Gilles Beaudoin, Denis Choquette, Boulos Haraoui, Hyman Tannenbaum, Joan M Meyer, John F Beary, Gary A Cline, et al. 2005. Long term evaluation of disease progression through the quantitative magnetic resonance imaging of symptomatic knee osteoarthritis patients: correlation with clinical symptoms and radiographic changes. *Arthritis research & therapy* 8(1):R21.

Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 779–788.

Revaud, Jerome, Matthijs Douze, and Cordelia Schmid. 2012. Correlation-based burstiness for logo retrieval. In *Proceedings of the 20th acm international conference on multimedia*, 965–968. ACM.

Richardson, Michael L., and Jonelle M. Petscavage. 2011. Verification Bias. *Academic Radiology* 18(11):1376–1381.

Riddle, Daniel L, Paul W Stratford, and Robert A Perera. 2016. The incident tibiofemoral osteoarthritis with rapid progression phenotype: development and

validation of a prognostic prediction rule. *Osteoarthritis and cartilage* 24(12):2100–2107.

Rippel, Oren, Jasper Snoek, and Ryan P. Adams. 2015. Spectral representations for convolutional neural networks. In *Proceedings of the 28th international conference on neural information processing systems - volume 2*, 2449–2457. NIPS'15, Cambridge, MA, USA: MIT Press.

Rota BulÃ², Samuel, Lorenzo Porzi, and Peter Kontschieder. 2018. In-place activated batchnorm for memory-optimized training of dnns. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* 115(3):211–252.

Sarraf, Saman, Ghassem Tofighi, and Disease Neuroimaging. 2016. DeepAD : Alzheimer ' s Disease Classification via Deep Convolutional Neural Networks using MRI and fMRI. *Biorxiv*.

Silverwood, V, M Blagojevic-Bucknall, C Jinks, JL Jordan, J Protheroe, and KP Jordan. 2015. Current evidence on risk factors for knee osteoarthritis in older adults: a systematic review and meta-analysis. *Osteoarthritis and cartilage* 23(4):507–515.

Simonyan, Karen, and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Arxiv eprint*. 1409.1556.

Smith, RW, P Egger, D Coggon, MID Cawley, and C Cooper. 1957. 9. kellgren jh, lawrence js. radiological assessment of osteoarthritis of the hip joint and acetabular dysplasia in osteoarthritis. *Ann Rheum Dis* 16:494–502.

Sun, Fangxuan, Jun Lin, and Zhongfeng Wang. 2016. Intra-layer nonuniform quantization for deep convolutional neural network. 1607.02720.

Suzuki, Kenji. 2017. Overview of deep learning in medical imaging. *Radiological physics and technology* 10(3):257–273.

Tolias, Giorgos, Ronan Sicre, and Hervé Jégou. 2016. Particular Object Retrieval With Integral Max-Pooling of CNN Activations. In *International Conference on Learning Representations*, 1–12.

Wang, Yan, Lingxi Xie, Chenxi Liu, Siyuan Qiao, Ya Zhang, Wenjun Zhang, Qi Tian, and Alan Yuille. 2017a. Sort: Second-order response transform for visual recognition. In *The ieee international conference on computer vision (iccv)*.

Wang, Yunhe, Chang Xu, Chao Xu, and Dacheng Tao. 2017b. Beyond filters: Compact feature map for portable deep model. In *ICML*, vol. 70 of *Proceedings of Machine Learning Research*, 3703–3711. PMLR.

Wen, Wei, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. 2016. Learning structured sparsity in deep neural networks. In *Advances in neural information processing systems 29*, ed. D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, 2074–2082. Curran Associates, Inc.

Wirth, Wolfgang, M-P Hellio Le Graverand, Bradley T Wyman, Susanne Maschek, Martin Hudelmaier, Wolfgang Hitzl, Michael Nevitt, Felix Eckstein, OAI Investigator Group, et al. 2009. Regional analysis of femorotibial cartilage loss in a subsample from the osteoarthritis initiative progression subcohort. *Osteoarthritis and cartilage* 17(3):291–297.

Wluka, Anita E, Stephen Stuckey, Judith Snaddon, and Flavia M Cicuttini. 2002. The determinants of change in tibial cartilage volume in osteoarthritic knees. *Arthritis & Rheumatism* 46(8):2065–2072.

Woloszynski, Tomasz, Pawel Podsiadlo, GW Stachowiak, M Kurzynski, LS Lohmander, and Martin Englund. 2012. Prediction of progression of radiographic knee osteoarthritis using tibial trabecular bone texture. *Arthritis & Rheumatism* 64(3):688–695.

Wu, Jiaxiang, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. 2016. Quantized convolutional neural networks for mobile devices. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Yoo, Tae Keun, Deok Won Kim, Soo Beom Choi, Ein Oh, and Jee Soo Park. 2016. Simple scoring system and artificial neural network for knee osteoarthritis risk prediction: a cross-sectional study. *PloS one* 11(2):e0148724.

Yuan, Zheng, and Debashis Ghosh. 2008. Combining multiple biomarker models in logistic regression. *Biometrics* 64(2):431–439.

Yue-Hei Ng, Joe, Fan Yang, and Larry S Davis. 2015. Exploiting local features from deep networks for image retrieval. In *Proceedings of the ieee conference on computer vision and pattern recognition workshops*, 53–61.

Zhang, Dongqing, Jiaolong Yang, Dongqiangzi Ye, and Gang Hua. 2018. Lq-nets: Learned quantization for highly accurate and compact deep neural networks. In *The european conference on computer vision (eccv)*.

Zhang, Junzhe, Sai Ho Yeung, Yao Shu, Bingsheng He, and Wei Wang. 2019. Efficient memory management for gpu-based deep learning systems. *arXiv preprint arXiv:1903.06631*.

Zhang, Weiya, Daniel F McWilliams, Sarah L Ingham, Sally A Doherty, Stella Muthuri, Kenneth R Muir, and Michael Doherty. 2011. Nottingham knee osteoarthritis risk prediction models. *Annals of the rheumatic diseases* 70(9):1599–1604.

Zhang, Xiangyu, Jianhua Zou, Xiang Ming, Kaiming He, and Jian Sun. 2015. Efficient and accurate approximations of nonlinear convolutional networks. In *The ieee conference on computer vision and pattern recognition (cvpr)*.

Zheng, Liang, Yi Yang, and Qi Tian. 2017. Sift meets cnn: A decade survey of instance retrieval. *IEEE transactions on pattern analysis and machine intelligence* 40(5): 1224–1244.

Zhou, Bolei, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 2921–2929.

Zhou, Zhaoye, Gengyan Zhao, Richard Kijowski, and Fang Liu. 2018. Deep Convolutional Neural Network for Segmentation of Knee Joint Anatomy. *Magn. Reson. Med.* doi:10.1002/mrm.27229.

Zhu, Bo, Jeremiah Z. Liu, Stephen F. Cauley, Bruce R. Rosen, and Matthew S. Rosen. 2018. Image reconstruction by domain-transform manifold learning. *Nature* 555(7697):487–492.