Automatic Hand Skeletal Shape Estimation From Radiographs

Radu Paul Mihail, Gongbo Liang, and Nathan Jacobs

***Abstract*— Rheumatoid arthritis (RA) is an autoimmune disease whose common manifestation involves the slow destruction of joint tissue, a damage that is visible in a radiograph. Over time, this damage causes pain and loss of functioning, which depends, to some extent, on the spatial deformation induced by the joint damage. Building an accurate model of the current deformation and predicting potential future deformations are the important components of treatment planning. Unfortunately, this is currently a time-consuming and labor-intensive manual process. To address this problem, we propose a fully automated approach for fitting a shape model to the long bones of the hand from a single radiograph. Critically, our shape model allows sufficient flexibility to be useful for patients in various stages of RA. Our approach uses a deep convolutional neural network to extract low-level features and a conditional random field (CRF) to support shape inference. Our approach is significantly more accurate than previous work that used hand-engineered features. We provide a comprehensive evaluation for various choices of network hyperparameters, as current best practices lack significantly in this domain. We evaluate the accuracy of our pipeline on two large datasets of hand radiographs and highlight the importance of the low-level features, the relative contribution of different potential functions in the CRF, and the accuracy of the final shape estimates. Our approach is nearly as accurate as a trained radiologist and, because it only requires a few seconds per radiograph, can be applied to large datasets to enable better modeling of disease progression.**

***Index Terms*— Rheumatoid arthritis, radiograph, conditional random field, convolutional neural network.**

I. Introduction

RHEUMATOID Arthritis (RA) is an autoimmune disease, with no known cure, that primarily affects synovial joints, especially those of the hands. The disease typically starts with inflammation and swelling, followed by the mutilation of joints through healthy tissue loss and scar tissue formation. The causes of the disease are multiple; genetic susceptibilities, lack of exercise, and environmental factors are considered to play a role in the onset and progression of the disease.

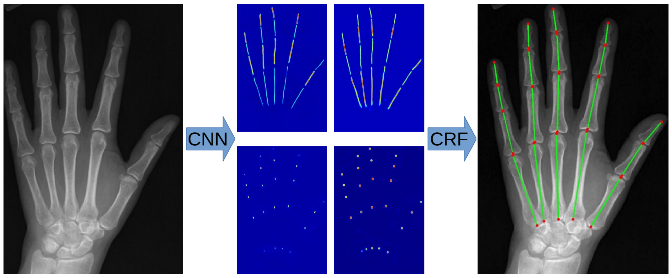

Fig. 1. Our radiograph shape fitting pipeline. Input image (left) is directly processed by a set of deep convolutional neural networks to produce feature maps (middle) that correspond to key anatomical features. The final shape (right) is inferred using a conditional random field (CRF).

Imaging of the hand is a routine procedure used by radiologists to assess the extent of the damage and to estimate the stage of disease progression. Disease staging can be a complex process, a function of key anatomical features and changes from a healthy baseline. Some of those features include the inter-joint spaces that tend to get smaller as the disease progresses. Recently, Pfeil *et al.* [20] have investigated using automated estimation of joint spaces as a predictor of RA disease progression with success. This motivates further work in this area, since different combination of medications may be more effective at earlier stages in the disease progression [22].

Automated joint space estimation is challenging for several reasons. First, the process for capturing radiographs, which are typically in the posteroanterior (PA) view, results in significant variability in hand placement and joint configuration. Such variability is acceptable for radiologists, but means that automated systems must cope with these differences. Second, appearance variations in radiographs appear due to inconsistent image acquisition parameters from calibration, digitization artifacts from film radiographs (scanner calibration), and anatomical differences. Appearance variations due to anatomy can be classified into two categories: differences from individual to individual and morphological differences due to disease (e.g., rheumatoid arthritis, osteoporosis), including inflammation and surgery.

We propose a fully automatic approach to estimating the configuration of the long bones of the hand from a radiograph. We adopt a previously introduced shape model [19] and combine bottom-up supervised feature extraction with a top-down shape inference process in the form of a conditional random field (CRF). We propose several novel improvements to the previous work on this problem, including improved low-level

Manuscript received March 27, 2019; accepted March 27, 2019. Date of publication April 12, 2019; date of current version June 28, 2019. This material is based upon work supported by the National Science Foundation under Grant No. IIS-1553116. *(Corresponding author: Radu Paul Mihail.)*

R. P. Mihail is with the Department of Computer Science, Valdosta State University, Valdosta, GA 31698 USA (e-mail: rpmihail@valdosta.edu).

G. Liang and N. Jacobs are with the Department of Computer Science, University of Kentucky, Lexington, KY 40506 USA (e-mail: gb.liang@uky.edu; nathan.jacobs@uky.edu).

Digital Object Identifier 10.1109/TNB.2019.2911026

features and better initialization for the CRF optimization. These changes substantially increase the accuracy and robustness without increasing the run time during inference.

We make the following key contributions: 1) an improved low-level feature extraction process that uses deep fully convolutional neural networks (CNNs), 2) an improved CRF initialization strategy, and 3) an evaluation on a large datasets that demonstrates the benefit of our improvements.

II. Related Work

Hand radiograph analysis has been investigated in the past and continues to receive attention from the computer vision and medical imaging research communities. We group related work into three categories: parametric shape model fitting to radiographs (most related to this work), hand radiograph pixel-level labeling algorithms and general radiograph analysis, a combination of parametric and non-parametric models and pixel-level operations. Our problem formulation is similar to face landmark localization and human pose estimation [4], [11], [24]. Recently, significant progress has been made in the above mentioned domains using end-to-end localization using CNNs. This is made possible by extremely large training sets, both real and synthetic. Our proposed method relies on significantly less training data, and thus a direct comparison is not made in this work.

Registration of hand bones has been explored using multiple imaging modalities. Simple radiographs are the least expensive modality and typically the first diagnostic imaging order by rheumatologists, for diagnosis as well as disease tracking [6], [17]. The carpal bones have overlap in 2D views and present significant challenges for vision based algorithms, but have been used with moderate success [3], [25]. Volumetric modalities such as computed tomography (CT or CAT) and magnetic resonance imaging (MRI) have also been used with registration algorithms, for both parametric and non-parametric model fitting. Chen *et al.* [3] propose a semantic segmentation and registration pipeline for the carpal bones from volumetric CT data using Grow Cut [25]. Mihail *et al.* [19] approximate these functions using dense SIFT [16] features and random decision forest (RDF) [5] classifiers. Since the relative size of hands (and bones) are similar in all images, the SIFT features at a fixed scale and orientation performed well with RDF classifiers. Our approach differs from others by not using manually engineered features. Instead, the features we use are learned using deep fully convolutional networks. Many fully convolutional network models have been proposed in the recent past, however, for biomedical image segmentation U-Net [21] is most similar to our design.

Parametric models for shape inference have been used in the past on hand radiographs. These algorithms typically rely on a set of image pre-processing or feature extraction steps for model initialization and inference. This work is closely related to that of Martín-Fernández *et al.* [18], who use a wire model whose wires correspond to the major axis of the hand's metacarpal and phalanges (long bones in the hand). Their method registers this wire model to previously unseen radiographs using a variation of thin plate splines (TPS) algorithm. However such representations are not well suited to modeling the joint displacements in RA patients.

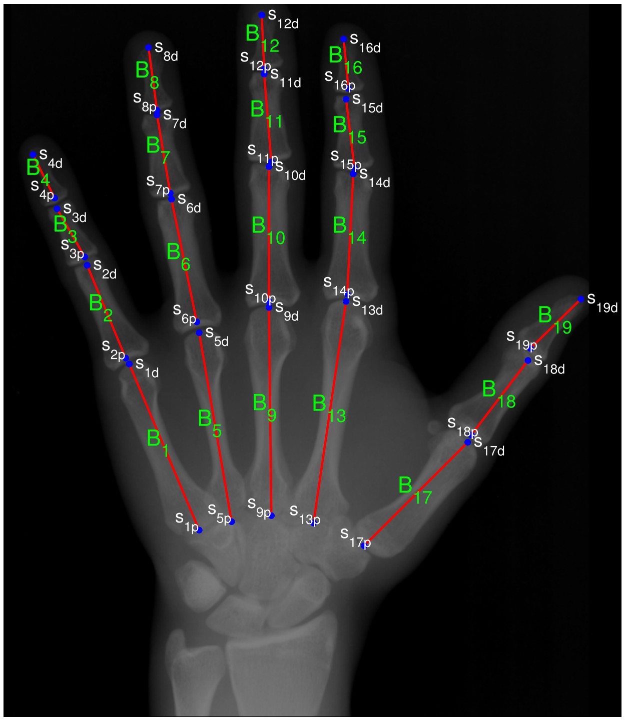

Fig. 2. Our shape model: each segment corresponds to a bone ($B_{1,\ldots,19}$). Individual points are indexed as proximal $s_{\{1,\ldots,19\}p}$ and distal $s_{\{1,\ldots,19\}d}$.

Pediatric skeletal maturity estimates are used to diagnose growth disorders, timing of surgical interventions, and endocrine disorders. This estimate is typically done on hand radiographs using Greulich and Pyle atlas (G&P) [9]. Bunch *et al.* [2] integrated automated bone age assessment methods to increase clinician reporting quality and speed. Larson *et al.* [15] directly use a deep learning architecture to estimate skeletal maturity from pediatric radiographs. Güraskin *et al.* [10] used hand radiographs to pediatric skeletal maturity using a cascade of morphological operations that result in six features used as input to classification algorithms (support vector machines, k-nearest neighbors, decision trees, and naïve Bayes).

In contrast to methods that directly estimate a property from hand radiographs, our method provides very accurate locations of the bones that are typically used in higher level analyses.

III. Problem Formulation

Given a roughly centered radiograph in posteroanterior (PA) view, our goal is to estimate the landmark points $s = \{s_1, \ldots, s_{19}\}$ that correspond to the epiphyses (end parts of long bones) of the metacarpal and phalanges along their main axis, as seen in Figure 2.

We approach the shape fitting problem by formulating the problem probabilistically. We define the best shape s for an image I, as the shape s that maximizes $P(s|I)$. We look for a factorization of the conditional $P(s|I)$ that is both tractable to approximate and captures conditional dependencies (e.g., shape points are not in the middle of a long bone). To accomplish this, we adopt a conditional random field (CRF) to represent the distribution. We integrate a set of low-level features, which are described below, in the potential functions of a CRF. The CRF has the following form:

$$P(s|I,\theta) = \frac{1}{Z} exp\{\sum_{i}\{\sum_{j} \Psi_j(s_{i_p}, s_{i_d}, \theta) + \sum_{k} \phi_k(s_i, \theta)\} + \zeta(s,\theta)\} \quad (1)$$

where i is an index over model segments, j and k index our binary and unary appearance terms, Z is the partition function, Ψ and ϕ are, respectively, binary and unary appearance terms, ζ is a shape model prior and θ is a weight vector we use to balance the various terms. In the following subsections, we define the potential functions, irrespective of the low-level features, discussed in detail in Section IV-A. In the remainder of this paper, we describe our approach to solving this problem, present evaluation results, and discuss the implications of this work.

IV. Approach

Our pipeline consists of two stages: low-level feature extraction, and a CRF whose maximum a posteriori (MAP) inference yields the optimal shape for a given image. In this section, we describe the various components of our approach for estimating the shape s for a given hand radiograph I, including: a deep convolutional neural network for extracting features from the imagery (Section IV-A); our process for converting these features into potential functions for our CRF (Section IV-B); our shape prior (Section IV-C); and the inference process, including shape initialization and CRF optimization (Section V).

A. Convnet for Hand Segmentation

For our low-level feature extraction process, we first define the following functions of image locations (p is a point on an image):

1) $f_j(p)$: a response function where the response is highest over epiphyseal surfaces of long bones aligned with their main axes,
2) $f_{pc}(p)$: a response function with maximal activation over proximal (closest to the center of the body) epiphyseal surfaces,
3) $f_{dc}(p)$: same as f_{pc} but with maximal responses at the distal (farthest from the center of the body) epiphyseal surfaces,
4) $f_{bc1}(p)$: a response function with maximal responses over the main axes of a subset of the long bones (metacarpals and middle phalanges), and
5) $f_{bc2}(p)$: a response function with maximal responses over the main axes of long bones (proximal and distal phalanges).

In the remainder of this section, we describe how we compute these functions from image data.

We formulate low-level feature extraction as a semantic segmentation problem. Neural networks have seen a tremendous amount of attention in the ML community over the last decade due to significant improvements in optimization algorithms and a better understanding of gradient and data propagation through the networks [12], [14], [23]. This formulation allows us to use existing knowledge from the semantic segmentation literature.

The biologically meaningful semantic classes we need are the joints and bones. Joint tissue is further divided into joint spaces, proximal and distal cortical surfaces. We group bones into two groups: 1) metacarpals and middle phalanges and 2) proximal and distal phalanges. Using this taxonomy, we propose to perform semantic segmentation using a novel fully convolutional network architecture.

Given an image I, and semantic label set $\mathcal{L} = \{l_1, l_2, \dots l_k\}$, a semantic segmentation algorithm will assign one of k labels to each pixel in I. Often, the output of such algorithms is a distribution over the labels, and the label with the highest likelihood is chosen. If there are two classes (e.g., background and bone tissue) $|\mathcal{L}| = 2$, then our previously defined response functions directly correspond to a pixel's segmentation likelihood. We now discuss our proposed convolutional network architecture.

Our network is designed using an encoder/decoder architecture. The encoder learns a low-dimensional representation of the input image. The decoder upsamples this low-dimensional representation to the original resolution, but whose content is semantically meaningful. This approach is similar to that of Badrinarayanan *et al.* [1] and Ronneberger *et al.* [21].

We are interested in 4 semantic classes: long bone tissue along its main axis, joint tissue (between long bones), proximal epiphyseal tissue (end of the bone nearest the body) and distal epiphyseal tissue (end of the bone farthest from the body).

Our encoder stage consists of a set of convolutional layers, batch normalization, and maxpool layers. The encoder stage ends at a bottleneck, where the network learns low resolution representations of the image. The encoder is composed of 5 convolutional layers (convolution, batch normalization, and maxpool) with 16 feature maps each. Each downsampling layer in the encoder has an upsampling pair layer in the decoder stage.

The decoder stage begins at the bottleneck, consists of upsampling layers, followed at last by a softmax classifier that makes the final pixel-wise semantic label prediction. The softmax layer has the same resolution as the input image. The upsampling layers use indices from corresponding maxpool layers in the encoder stage. Each convolutional layer is followed by a batch normalization layer and a rectified linear activation (ReLU) layer. In Table I we summarize the network's convolutional layer hyperparameters during training.

B. CRF Potential Functions

The low-level features are combined in various ways to construct a set of binary and unary potential functions that our CRF optimization process uses to fit the final shape.

1) Unary Potential Functions: The unary potential functions encode the compatibility between individual shape points s_{i_p}

TABLE I

OUR NETWORK ARCHITECTURE: b IS THE BATCH SIZE AND c IS THE NUMBER OF SEMANTIC SEGMENTATION CLASSES

Layer	Shape	Kernel
$conv0$	$b \times 16 \times 600 \times 460$	7×7
$pool0$	$b \times 16 \times 300 \times 230$	3×3
$conv1$	$b \times 16 \times 300 \times 230$	7×7
$pool1$	$b \times 16 \times 150 \times 115$	3×3
$conv2$	$b \times 16 \times 150 \times 115$	7×7
$pool2$	$b \times 16 \times 75 \times 57$	3×3
$conv3$	$b \times 16 \times 75 \times 57$	7×7
$pool3$	$b \times 16 \times 37 \times 28$	3×3
$conv4$	$b \times 16 \times 37 \times 28$	7×7
$pool4$	$b \times 16 \times 18 \times 14$	3×3
$upsample4$	$b \times 16 \times 37 \times 28$	3×3
$conv4\_D$	$b \times 16 \times 37 \times 28$	7×7
$upsample3$	$b \times 16 \times 75 \times 57$	3×3
$conv3\_D$	$b \times 16 \times 75 \times 57$	7×7
$upsample2$	$b \times 16 \times 150 \times 115$	3×3
$conv2\_D$	$b \times 16 \times 150 \times 115$	7×7
$upsample1$	$b \times 16 \times 300 \times 230$	3×3
$conv1\_D$	$b \times 16 \times 300 \times 230$	7×7
$upsample0$	$b \times 16 \times 600 \times 460$	3×3
$conv0\_D$	$b \times 16 \times 600 \times 460$	7×7
$conv\_00$	$b \times 16 \times 600 \times 460$	9×9
$conv\_classifier$	$b \times c \times 600 \times 460$	3×3

(proximal), s_{i_d} (distal) and the image, where $i \in \{1, \dots, 19\}$. The first potential function we define encourages shape points to be on or near cortical surfaces of bones (both proximally and distally). This function will make use of the low-level feature function f_j through a function $df_j(p)$ (shown in Figure 3), defined as follows:

$$df_j = \begin{cases} 1 - f_j(p), & \text{if } f_j(p) \text{ is non-zero} \\ d(p, f_j) & \text{otherwise} \end{cases}$$

where $d(p, f)$ is the distance to the closest non-zero point in f from p.

We can now define the unary potential ϕ_1 as follows:

$$\phi_1(s_i) = \theta_1 df_j(s_i)$$

We further define two unary potential functions that encourage shape points to discriminate between proximal and distal cortical surfaces. These functions increase the accuracy of the final fit by encouraging an image-driven joint distance.

Intuitively, the functions ϕ_1 provides a rough alignment with joints, however, it does not discriminate between the proximal and distal bone in a joint. We make use of low-level feature functions f_{dc} and f_{pc} to define augmented distance functions df_{pc} and df_{dc} as follows:

$$df_{pc} = \begin{cases} 1 - f_{pc}(p), & \text{if } f_{pc}(p) \text{ is non-zero} \\ d(p, f_{pc}) & \text{otherwise} \end{cases}$$

and

$$df_{dc} = \begin{cases} 1 - f_{dc}(p), & \text{if } f_{dc}(p) \text{ is non-zero} \\ d(p, f_{dc}) & \text{otherwise} \end{cases}$$

where $d(p, f)$ is the same distance function to a non-zero point p in f as in Equation 2.

Our potential functions are defined as follows:

$$\phi_2(s_{i_d}) = \theta_2 df_{dc}(s_{i_d})$$

and

$$\phi_3(s_{i_p}) = \theta_3 df_{pc}(s_{i_p})$$

2) Binary Potential Functions:

The potential functions Ψ_1 and Ψ_2 encode the compatibility between a shape segment i, with points i_p (proximal), i_d (distal) and bone tissue using f_{bc1} and f_{bc2} (f_{bc_x}). These functions should be at a minima when any segment s is directly over the main axis of a bone. We define an augmented distance function $df_{b_x}(p)$ from $f_{bc_x}(p)$ as follows:

$$df_{b_x} = \begin{cases} 1 - fbc_x(p), & \text{if } f_{bc_x}(p) \text{ is non-zero} \\ d(p, f_{bc_x}) & \text{otherwise} \end{cases}$$

If we ensure $\max(f_{bc_x}) \leq 1$ and interpret it as the probability of pixel's p membership to the long bone semantic category, the complement of the collection of non-zero points from f_{bc_x} adds precision to the basins of attraction. Having df_{b_x} defined (shown in Figure 3), we can now define Ψ_1 as follows:

$$\Psi_1(s_{i_p}, s_{i_d}) = \theta_4 \frac{1}{n} \sum_{n=1}^{t} df_{b_1}(p_{x_n}, p_{y_n})$$

and

$$\Psi_2(s_{i_p}, s_{i_d}) = \theta_5 \frac{1}{n} \sum_{n=1}^{t} df_{b_2}(p_{x_n}, p_{y_n})$$

C. Hand Shape Potential Function

The shape prior term $\zeta(s, \theta)$ is a global term used to encourage inferred shapes to lie closely in the subspace of known shapes. We model this knowledge using Probabilistic Principal Component analysis, where the goal is to relate shape s to a k-dimensional vector x, where $k \ll \dim(s)$ and x is of zero mean and unit covariance ($I(k)$), such that:

$$s^T = Wx^T + \bar{s} + \epsilon$$

where $\bar{s}$ is the average shape and ϵ is a normally distributed model noise component. Using this model, we can assign a probability to any shape s:

$$P(s) = \mathcal{N}(\bar{s}, WW^T + \sigma^2 I(k))$$

and formulate our prior as a weighted negative log likelihood of $P(s)$:

$$\zeta(s, \theta) = \theta_6(-log(P(s)))$$

D. Shape Inference

To perform shape inference, we compute the MAP solution to the CRF by taking the log-likelihood of the model, yielding an energy function E:

$$E(\hat{s}, \theta, I) = \arg\min_s \sum_i \{\sum_j \Psi_j(s_{i_p}, s_{i_d}, \theta) + \sum_k \phi_k(s_i, \theta)\} + \zeta(s, \theta). \quad (2)$$

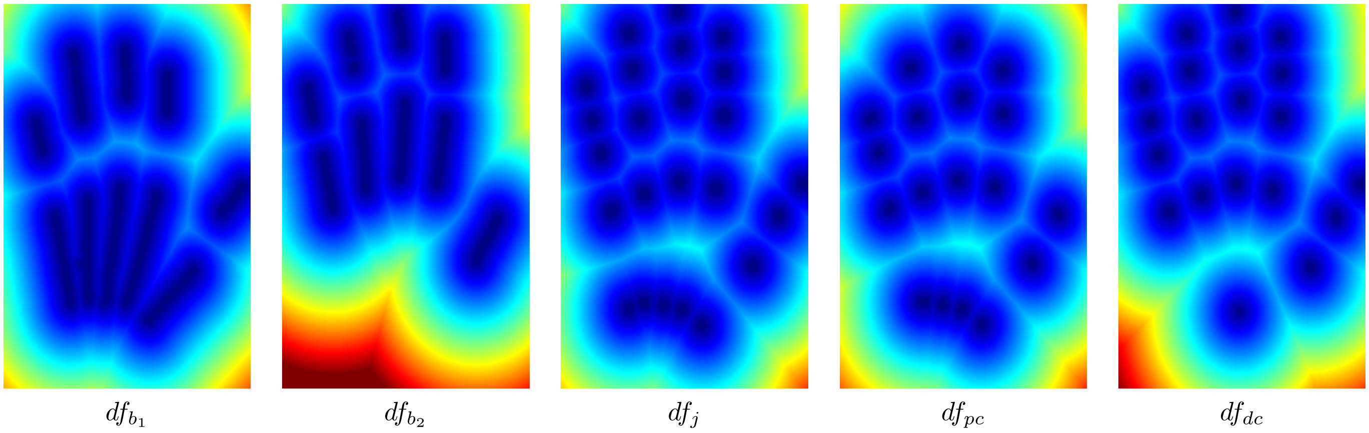
df_{b_1} df_{b_2} df_j df_{pc} df_{dc}

Fig. 3. Visualization of df_{b_1}, df_{b_2}, d_j, df_{pc} and df_{dc}.

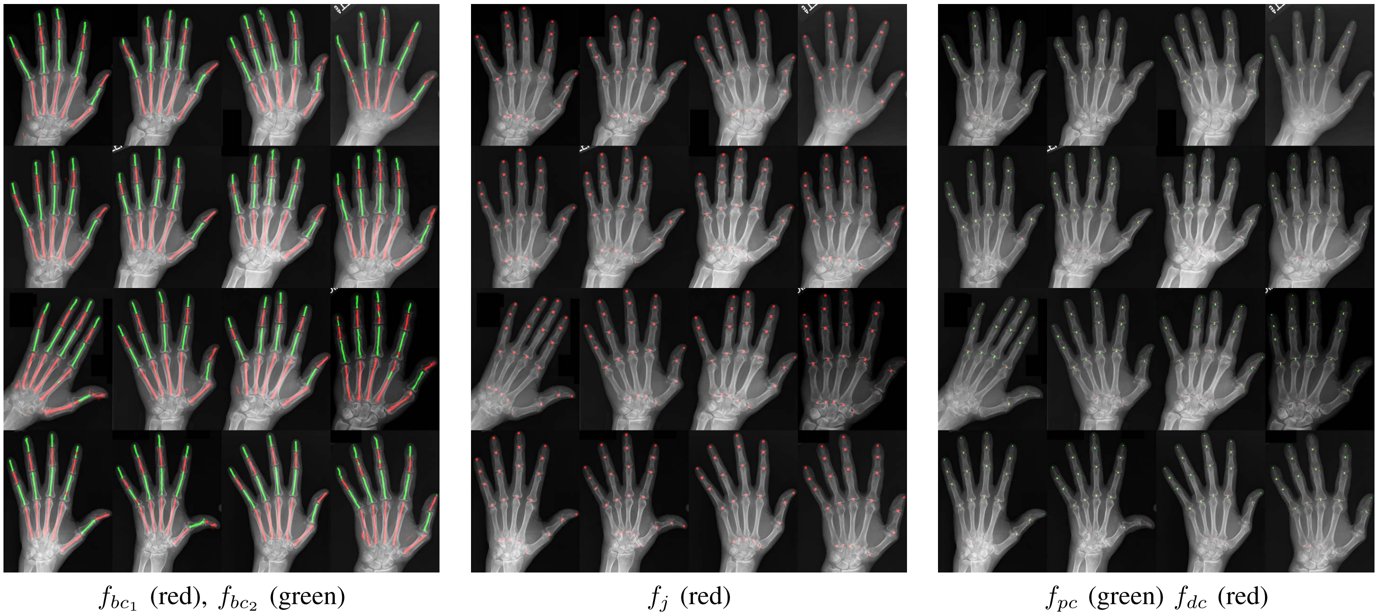
f_{bc_1} (red), f_{bc_2} (green) f_j (red) f_{pc} (green) f_{dc} (red)

Fig. 4. Sample responses f_{bc_1}, f_{bc_2}, f_j, f_{pc} and f_{dc}. Best seen in color, digital format.

As we can see from Equation 2, the optimal shape $\hat{s}$ for image I using model parameters θ is obtained when the energy function is minimized. The minimization problem is non-convex with many possible local minima, hence, a robust initialization procedure is important. The initialization routine computes a shape $\hat{s}$ from which local search is started.

1) Shape Initialization: Our initialization process consists of two stages: rough initialization using a variation of the Iterative Closest Point (ICP) algorithm and a fine-tuning local search stage using PCA. For the rough initialization stage, we compute the connected component statistics (centroid and major spanning axis) for our bone tissue responses, f_{bc_1} and f_{bc_2}. These segments are used to register an average hand with the ICP algorithm.

In the fine-tuning stage of our initialization process, we compute the PCA basis of hand shapes in our training set. Let W be the PCA basis and x be coefficients, such that a shape s can be reconstructed from coefficients as follows:

$$s = W^T x + \bar{s}$$

where $\bar{s}$ is the shape average. Given the joint center response centroids J and bone tissue centroids B, we define the following minimization problem:

$$E(\hat{s}, x, I) = \arg\min_{x} \sum_{i,j} ||J_i - \hat{s}_j||_2^2 + \sum_{k,l} ||B_k - 0.5(\hat{s}_l + \hat{s}_{l+1})||_2^2 + \sum |x| \quad (3)$$

where i is the index of the joint centroid nearest to shape point s_j, and k is the index of the bone centroid nearest to the average of a segment $s_l s_{l+1}$, and the last term is a regularization term that encourages the inferred shape to be close to the average shape.

V. Evaluation

We evaluated our approach on two datasets and found that our method significantly increases the accuracy for joint location estimation relative to the previous work. This result demonstrates that even with relatively small datasets it is

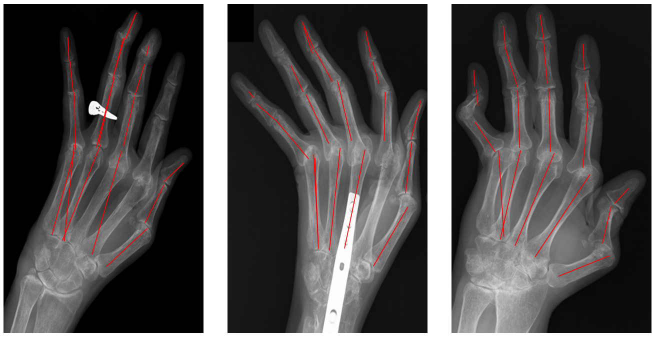

Fig. 5. Shapes with highest error in the RA dataset.

possible to use learned feature representations to achieve better performance than hand engineered features and shallow learning techniques.

A. Datasets

We use two datasets, the Digital Hand Atlas Database[1] and a set of 116 radiographs of RA patients from the University of Kentucky Department of Radiology. The University of Kentucky dataset contains additional annotations of rheumatoid arthritis disease stage. Both datasets have been manually annotated and verified by a radiologist. We use these ground truth shapes to train the convnets, estimate CRF model parameters and quantitatively evaluate inferred shapes and the low-level features used in the CRF inference process. Since the data is similar in both datasets, we alternate using one dataset for training and the other for testing (e.g., we trained on UK and report test on Hand Atlas, and vice-versa).

We augmented both datasets by rotating each image in 10 degree increments from -30 deg to $+30$ deg while maintaining the resolution. This was done to learn invariance to rotation and resulted in a significant improvement in robustness. For evaluation, we look at two aspects of our approach: 1) the estimated PDM and 2) the performance of our low-level feature estimators.

B. Implementation Details

The CNNs were implemented using a customized version of Caffe [13]. The resolution of the input and output feature maps was 460×600 in our experiments. Our networks were trained in CPU on a 40-core Intel blade server using a mini-batch size of 4. All models were trained on one of the datasets and tested on the other.

There are known trade-offs between the filter size and network depth in convolutional layers. We obtained the best performance with 7×7 kernels in the convolutional layers. During the CNN architecture development, we also noted a significant performance improvement from the addition of batch normalization layers.

Our approach to network design was incremental. We started with a simple encoder-decoder architecture, with a depth of 2 layers for each stage and a fixed kernel size of 7×7. We gained accuracy increasing the number of layers to 5. A depth of more than 5 layers per stage decreased accuracy. While there is increasing evidence of a relationship between kernel size and network depth, we obtained the best results with the structure we present in this paper.

C. Network Design Choices for Low-Level Feature Extraction

In this section, we systematically evaluate the hyperparameters used in the encoder/decoder networks. Three sets of tests were performed by altering depth (i.e., number of layers), kernel size, and number of kernels. For each design choice, we also controlled the number of output feature maps, resulting in two extra categories of models: single output models (one model per low-level feature, named as $Model_{1M1F}$), and multi-output models (one model to predict all low-level features, $Model_{1M5F}$). We trained and tested a total of 222 models with different hyperparameters choices on the same train/test sets. Each model was trained for 100 epochs. Every experiment was repeated three times. The results presented in this section are the averaged results over the three times.

1) Depth and Kernel Size: We use an 11-layer encoder/decoder network to extract the low-level feature in this study, which has five convolution layers before the bottleneck and five convolution layers after the bottleneck layer. Each layer has 16 7×7 kernels.

According to our experiments, this was the optimal choice of hyperparameters. In order to gain more insight into design choices, we first evaluated how the depth (the number of layers before and after the bottleneck) and kernel size affect the model performance. We tested six depths, $Depth = 2$ to 7 (there are 2 to 7 convolution layers before and after the bottleneck) and two kernel sizes, $Kernel = 7 \times 7$ or 3×3.

Our experiments show that 7×7 kernels generally perform better than 3×3 kernels. The exception is the f_{pc} feature of both of $Model_{1M5F}$ and $Model_{1M1F}$. Among all the models, $Kernel = 7 \times 7$, $Depth = 5$ has consistently better performance. $Model_{1M5F}$ and $Model_{1M1F}$, both with $Kernel = 7\times7$ and $Depth = 5$, achieved the best performance on 3 out of 4 features. See Table III for details.

2) Fixed Number of Kernels: The model used in this study has 16 kernels in each layer. We evaluated whether this is the optimal choice. In this experiment, we fix the depth and kernel size at 5 and 7 but use different numbers of kernels. In these models, each layer has the same number of kernels. We evaluated models with 2, 4, 8, 16, and 32 kernels per layer.

This experiment shows that for most of the models, a larger number of kernels results in better performance. For instance models of 2 and 4 kernels at each layer constantly performed worse than the others. Models with 16 and 32 kernels per layer achieved the best results. See Table IV for details.

3) Exponentially Increased Number of Kernels: U-net, one of the popular encoder/decoder architecture, increases the number of kernels at each layer exponentially with a factor of 2. For instance, if the encoder starts with k kernels at the first convolutional layer, the number of the kernel of the second layer equals $k \times 2^1$, the number of the kernel of

[1] http://ipilab.usc.edu/BAAweb/

TABLE II
LOW-LEVEL FEATURE QUANTITATIVE EVALUATION (CNN/RDF)

	f_{bc_1}&f_{bc_2}	f_j	f_{pc}	f_{dc}
Average Positive Probability	**0.6223** / 0.5069	**0.7128** / 0.4301	**0.4830** / 0.3263	**0.5960** / 0.3669
MCR (% of image)	**0.0319** / 0.1297	**0.0071** / 0.0252	**0.0021** / 0.0137	**0.0022** / 0.0088

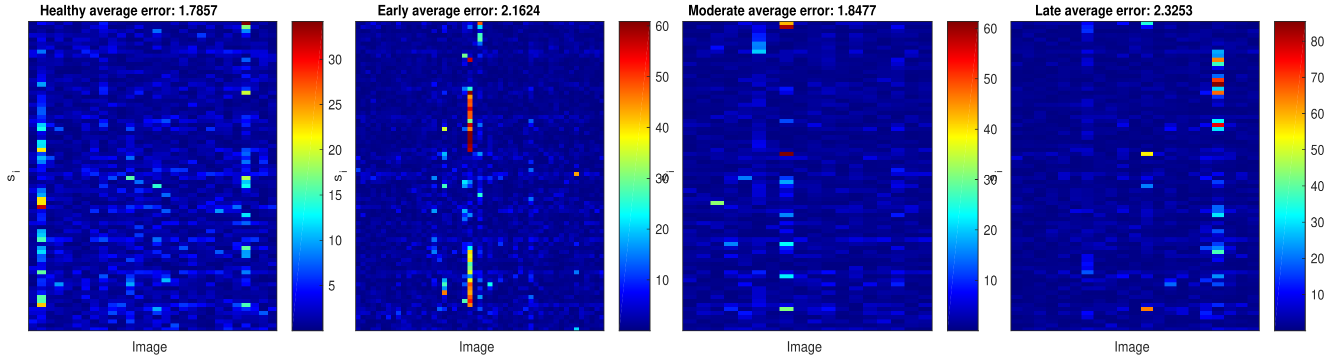

Fig. 6. RA dataset errors computed as sum of absolute differences from ground truth for healthy, early, moderate and late stage. All models have been trained on the Hand Atlas dataset.

TABLE III
TESTING RESULTS FOR DIFFERENT DEPTHS AND KERNEL SIZES

Depth	Kernel	**Model**$_{1M5F}$								**Model**$_{1M1F}$							
		f_{bc1}&f_{bc2}		f_j		f_{dc}		f_{pc}		f_{bc1}&f_{bc2}		f_j		f_{dc}		f_{pc}	
		MCR[1]	APP[2]	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP
2	7	0.028	0.67	0.0128	0.75	**0.0016**	0.46	0.0021	0.64	0.0305	0.72	0.01	0.80	**0.0015**	0.60	**0.0019**	0.67
3	7	0.0212	0.82	0.0102	0.82	0.0022	0.63	0.0035	0.77	0.0218	0.83	0.0075	0.80	**0.0015**	0.62	0.0022	**0.79**
4	7	0.0197	0.85	0.0098	0.82	0.0028	0.65	0.0034	0.74	0.0162	0.84	0.0075	0.81	0.0016	0.63	0.002	0.71
5	7	**0.0174**	**0.86**	0.0096	0.83	**0.0016**	**0.68**	**0.0023**	**0.75**	**0.0151**	**0.86**	0.0074	**0.85**	**0.0015**	0.68	**0.0019**	0.74
6	7	0.019	0.84	0.0095	0.82	0.0022	0.65	0.0028	0.72	0.0157	0.84	0.0078	0.82	0.0015	**0.70**	0.0022	0.78
7	7	0.0186	0.84	**0.0088**	**0.84**	0.0019	0.64	0.0023	0.71	0.0155	0.83	**0.0072**	0.81	**0.0015**	0.69	**0.0019**	0.73
2	3	0.028	0.48	0.0222	0.65	**0.0016**	0.45	0.0019	0.49	0.0188	0.55	0.0123	0.65	**0.0015**	0.43	**0.0018**	0.59
3	3	0.0211	0.64	0.0141	0.71	0.0017	0.52	0.0026	0.57	0.0224	0.68	0.093	0.79	**0.0015**	0.53	**0.0018**	0.67
4	3	0.0269	0.79	**0.0095**	0.78	**0.0016**	0.58	**0.0018**	0.71	0.0184	0.80	0.0093	0.76	0.0016	0.58	**0.0018**	0.68
5	3	0.0197	0.81	0.0102	**0.81**	**0.0016**	0.61	0.0019	**0.76**	0.0165	0.78	0.0077	0.81	**0.0015**	0.62	**0.0018**	0.71
6	3	**0.0186**	0.81	0.0101	**0.81**	**0.0016**	**0.68**	0.0019	0.72	**0.0159**	**0.82**	**0.0074**	**0.82**	0.0016	**0.67**	**0.0018**	**0.79**
7	3	0.0189	**0.85**	0.0102	**0.81**	0.0019	0.64	0.0019	0.75	0.0162	0.81	0.0076	**0.82**	0.0016	0.66	0.0019	0.73

[1] MCR (% of image), [2] Average Positive Probability

TABLE IV
TESTING RESULTS FOR DIFFERENT NUMBER OF KERNEL, THE NUMBER OF KERNEL IS SAME FOR EVERY LAYER WITHIN THE SAME MODEL

# of kernels	**Model**$_{1M5F}$								**Model**$_{1M1F}$							
	f_{bc1}&f_{bc2}		f_j		f_{dc}		f_{pc}		f_{bc1}&f_{bc2}		f_j		f_{dc}		f_{pc}	
	MCR[1]	APP[2]	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP
2	0.051	0.73	0.041	0.63	**0.0016**	0.31	**0.0018**	0.41	0.0287	0.82	0.0129	0.55	0.0026	0.60	0.0041	0.71
4	0.0235	0.84	0.0136	0.80	0.0045	0.56	0.0042	0.65	0.0186	0.84	0.0094	0.83	0.0018	0.65	0.0024	0.72
8	0.00195	0.85	0.0091	0.81	**0.0016**	0.64	0.0021	0.72	0.0159	0.84	0.0074	**0.86**	0.0017	**0.69**	**0.0019**	0.71
16	**0.00174**	**0.86**	0.0096	**0.83**	**0.0016**	**0.68**	0.0023	**0.75**	**0.0151**	**0.86**	0.0074	0.85	**0.0015**	0.68	**0.0019**	**0.74**
32	**0.00174**	0.85	**0.0082**	0.81	0.0023	0.62	0.0025	0.73	0.0153	0.84	**0.0067**	0.82	**0.0015**	0.62	0.0021	0.72

[1] MCR (% of image), [2] Average Positive Probability

the third layer equals to $k \times 2^2$, and so on. In this experiment, we evaluate whether this type of architecture has an effect on the model performance. Since U-net uses a factor of 2 and we already know the performance of U-net, in this work, we only evaluated the exponential between 1 and 2, more specifically, we evaluated 1.2, 1.4, 1.6, and 1.8. We set the number of kernels at the first layer to 2, 4, 8, 16, and 32.

We found that a larger number of kernels at the initial layer and a larger exponential number is associated with better performance until the number of kernels reaches 16, after which the model performance decreases significantly. See Table V for more detail.

D. CRF Model Parameter Selection

1) Model Parameter Optimization: The parameter set θ from Equation 2 is done using standard model selection, as they encode the relative weight of each feature in the shape inference process. The optimal θ are found by minimizing the

TABLE V

TESTING RESULTS FOR DIFFERENT NUMBER OF KERNEL, THE NUMBER OF KERNEL IS CHANGED EXPONENTIALLY

Exponential	**# of kernels**[1]	**Model**$_{1M5F}$								**Model**$_{1M1F}$							
		$f_{bc1}\&f_{bc2}$		f_j		f_{dc}		f_{pc}		$f_{bc1}\&f_{bc2}$		f_j		f_{dc}		f_{pc}	
		MCR[2]	APP[3]	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP	MCR	APP
1.2	2	0.0365	0.72	0.0096	0.51	**0.0015**	0.21	**0.0018**	0.24	0.0233	0.84	0.0128	0.76	0.0015	0.22	0.0037	0.71
1.2	4	0.0174	0.86	0.0104	0.83	0.0032	0.63	0.0037	0.75	0.0171	0.84	**0.0075**	0.78	0.0015	0.64	**0.002**	0.71
1.2	8	**0.0163**	**0.87**	0.0093	**0.88**	**0.0015**	**0.73**	**0.0018**	**0.81**	**0.0158**	**0.86**	0.0079	**0.81**	0.0015	0.59	0.0023	**0.73**
1.2	16	0.0165	**0.87**	**0.0083**	0.86	**0.0015**	0.71	**0.0018**	0.78	0.016	0.82	**0.0075**	0.78	0.0015	0.61	0.0024	0.72
1.2	32	0.0171	0.86	0.087	0.84	0.0016	0.65	**0.0018**	0.75	0.0204	**0.86**	0.0102	0.76	0.0015	**0.77**	0.0025	**0.73**
1.4	2	0.0215	0.83	0.0095	0.81	**0.0015**	0.57	0.0033	0.68	0.0191	0.71	0.0094	0.79	**0.0015**	0.6058	0.0021	0.51
1.4	4	**0.0162**	0.87	0.0094	**0.84**	**0.0015**	**0.71**	0.0025	**0.81**	**0.0157**	0.84	0.0082	**0.88**	0.0017	0.63	0.0024	0.71
1.4	8	**0.0162**	**0.88**	0.0094	**0.84**	**0.0015**	0.68	**0.0018**	0.76	0.0161	0.85	0.0079	0.82	0.0016	0.59	**0.0019**	**0.73**
1.4	16	0.0166	0.87	**0.0078**	**0.84**	0.0016	0.70	**0.0018**	0.77	0.0167	0.83	**0.0077**	0.85	0.0016	0.62	**0.0019**	**0.73**
1.4	32	0.0174	0.85	0.0087	**0.84**	**0.0015**	0.67	0.0021	0.76	0.0199	**0.86**	0.0094	0.77	0.0016	**0.75**	0.0029	0.69
1.6	2	0.0169	0.87	0.0101	0.82	0.0046	0.56	0.0047	0.66	0.0162	0.70	0.0082	0.82	0.0017	0.53	0.0019	0.73
1.6	4	0.0172	**0.92**	0.0087	0.85	0.0019	0.69	0.0025	**0.82**	**0.0147**	**0.84**	**0.071**	0.80	0.0016	0.68	0.002	0.74
1.6	8	0.0169	0.86	0.0083	0.84	0.0016	0.69	0.0019	0.72	0.0158	0.81	0.077	**0.83**	**0.0015**	0.62	0.002	**0.75**
1.6	16	**0.0154**	**0.92**	**0.0064**	**0.88**	**0.0015**	**0.74**	**0.0018**	0.78	0.0169	0.81	**0.071**	0.81	**0.0015**	0.68	**0.0018**	**0.75**
1.6	32	0.0288	0.83	0.0098	0.81	0.0022	0.56	0.0026	0.70	0.0186	**0.84**	0.079	0.80	0.0022	**0.70**	**0.0018**	0.70
1.8	2	**0.0151**	0.87	0.0076	0.85	0.0018	0.70	0.0021	0.76	**0.0155**	0.71	0.072	0.80	**0.0015**	0.58	0.0021	0.70
1.8	4	0.0171	**0.89**	0.0091	0.83	**0.0015**	0.68	0.0022	0.75	**0.0155**	0.85	**0.071**	0.80	0.0016	0.65	**0.0018**	**0.74**
1.8	8	0.0166	0.87	0.008	0.88	**0.0015**	0.72	0.0019	0.77	0.0163	0.82	0.078	0.83	0.0016	0.60	**0.0018**	0.70
1.8	16	0.0155	**0.89**	**0.0068**	**0.90**	**0.0015**	**0.81**	**0.0018**	**0.79**	0.0171	0.84	0.074	**0.84**	**0.0015**	0.69	**0.0018**	**0.74**
1.8	32	0.027	0.80	0.0114	0.75	0.0016	0.56	**0.0018**	0.64	0.0205	**0.86**	0.079	0.81	0.0016	**0.71**	**0.0018**	0.71

[1] # of kernels at the first layer, [2] MCR (% of image), [3] Average Positive Probability

global pixel-wise error for the entire training set, as follows:

$$\hat{\theta}^{ML} = \arg\min_{\theta} \sum_{i} \left\| \arg\min_{s} E(s, \theta, I_i) - s_i^{GT} \right\|_2^2 \qquad (4)$$

The above minimization is non-convex, hence many local minima are possible. We use the simplex method to find the optimal model parameters.

E. Inference

1) Shape Inference Evaluation: We evaluate our model by computing the sum of absolute differences between ground truth shapes and our model inference results. Shape inference is done by minimizing Equation 2. In practice, we use a coordinate descent algorithm, with a search region per coordinate is 7 pixels. This approach converges in a few seconds on a modern desktop multi-core CPU. This algorithm was implemented in MATLAB.

2) Ablation Study: To gain insight into the CRF model functions, we perform an ablation study by removing each potential function from the model, and noting the increase in error. In Table VI, we show the results. All terms are shown to be helpful. Removing joint center information contributes to the highest increase in error. The hand prior term is also significant, since its removal causes the second most significant increase in error.

F. Qualitative and Quantitative Comparisons

By visual inspection, we found the CNN features to be far superior to DSIFT+RDF. In Figure 7, we show samples of each feature function $f_j(p)$, $f_{dc}(p)$, $f_{pc}(p)$,$f_{bc_1}(p)$ for an image using both CNN and DSIFT+RDF. Visually, one can conclude that CNN features are much cleaner, with less noise and increased localization accuracy.

TABLE VI

INCREASE IN ERROR (IN PIXELS) WHEN REMOVING INDIVIDUAL CRF TERMS. THE TERM THAT CONTRIBUTES TO THE HIGHEST ERROR INCREASE IS THE JOINT CENTER TERM ϕ_1, FOLLOWED BY THE PRIOR ζ

Term	Ψ_1	Ψ_2	ϕ_2	ϕ_3	ϕ_1	ζ
Error increase	2.148	2.296	1.540	1.941	2.950	2.406

To quantitatively evaluate the features, we look at two performance metrics, summarized in Table II. First, for f_{bc_1} and f_{bc_2}, we take all the pixels along segments $s_{1..19}$ and compare the average probability. This number should be as close to 1 as possible, the higher the better, since responses along the main bone axes should activate. CNN features are roughly 12% more confident. We take the same approach for the other classifiers. CNN exceeds DSIFT+RDF in all cases. As a second metric, we compute the ratio of incorrectly classified pixels, specifically the misclassification rate (MCR), computed as the ratio of incorrectly classified pixels and the total number of pixels in the image (thresholding responses at ≥ 0.25). This metric gives us an insight into the localization precision of the methods. Using this metric, our proposed CNN-based feature extraction method outperforms previous results based on DSIFT+RDF for all instances.

In Figure 6, we show the errors, in pixels, on the RA dataset. The overall average error for all stages is 2.0521 pixels compared to 3.0333 pixels in previous work [19], a significant improvement. In Figure 5 we show the hands that had the highest fitting errors in the RA dataset. The overall error for

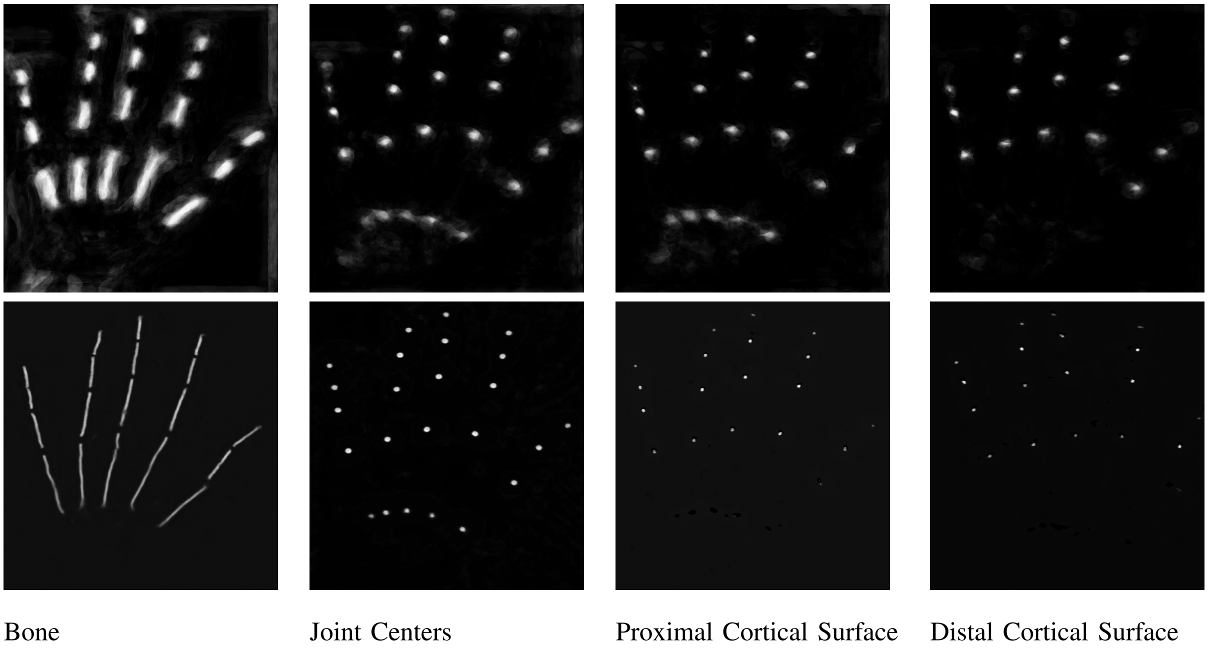

Fig. 7. Top: DSIFT+RDF features. Bottom: convnet features. The difference in quality and precision is evident by simple visual inspection.

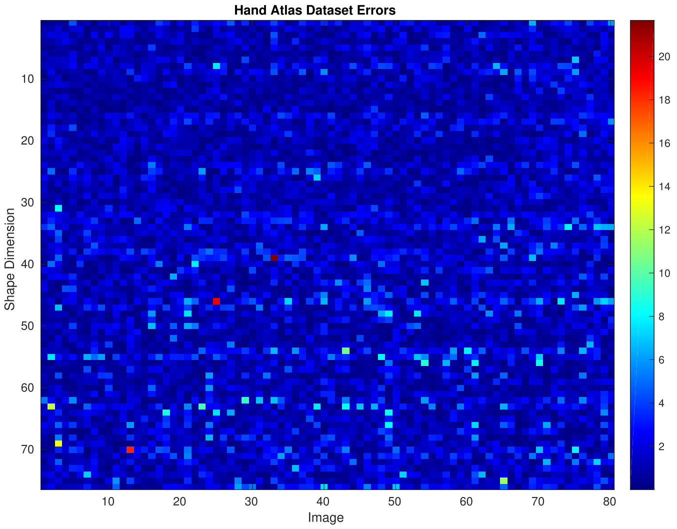

Fig. 8. Hand Atlas dataset errors computed as sum of absolute differences from ground truth. All models have been trained on the UK dataset.

the Hand Atlas dataset is 1.46 compared to 2.72 in Mihail *et al.* [19], on the same images with the same resolution.

VI. Conclusion and Future Work

Simple hand radiographs are used in clinical practice for skeletal development assessment, rheumatic disease progression estimation, among other uses. Automatic estimation of joint spaces and long-bone segments is useful in modeling disease progression, e.g., RA. This paper improves on previous work that fits a shape model to hand radiographs. We propose a hand shape inference method from radiographs based on deep convolutional neural network features. The CNN features are superior to those obtained using shallow features, e.g., SIFT [16]. We quantitatively evaluate the features standalone, as well as compare the final shape inference results using shallow and deep CNN features. We note an average improvement in the shape inference of roughly 1 pixel per shape dimension compared to the previous work.

The task of shape estimation from hand radiographs is related to non-medical shape estimation tasks, for example the task of facial landmark localization [7], [8], [26]. Recent approaches for this task, and other similar tasks, propose deep learning architectures that are suitable for end-to-end optimization. Our task is potentially well suited for end-to-end optimization, but existing sets of hand radiographs are significantly smaller than those available for other domains. This leads to significant problems with overfitting, which motivated our hybrid approach of using deep learning for low-level features but CRF optimization for top-down inference. We expect that by combining our proposed techniques with larger datasets and end-to-end learning we will be able to make further advances in future work. Upon acceptance of this paper, we will release our fully trained models as well as source code that implements CRF optimization.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. (2015). "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." [Online]. Available: https://arxiv.org/abs/1511.00561

[2] P. M. Bunch, T. A. Altes, J. McIlhenny, J. Patrie, and C. M. Gaskin, "Skeletal development of the hand and wrist: Digital bone age companion—a suitable alternative to the Greulich and Pyle atlas for bone age assessment?" *Skeletal Radiol.*, vol. 46, no. 6, pp. 785–793, Jun. 2017.

[3] X. Chen, J. Graham, and C. Hutchinson, "Integrated frameworkfor simultaneous segmentation and registration of carpal bones," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 433–436.

[4] X. Chen, E. Zhou, Y. Mo, J. Liu, and Z. Cao, "Delving deep into coarse-to-fine framework for facial landmark localization," in *Proc. CVPRW*, Jul. 2017, pp. 2088–2095.

[5] A. Criminisi, J. Shotton, and E. Konukoglu, “Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning,” Microsoft Research Cambridge, Cambridge, U.K. Tech. Rep. MSRTR-2011-114, 2011, vol. 5, no. 6, p. 12.

[6] K. D. Deane, J. M. Norris, and V. M. Holers, “Preclinical rheumatoid arthritis: Identification, evaluation, and future directions for investigation,” *Rheumatic Disease Clinics North Amer.*, vol. 36, no. 2, pp. 213–241, May 2010.

[7] Z. Deng, K. Li, Q. Zhao, and H. A. Chen, “Face landmark localization using a single deep network,” in *Proc. Chin. Conf. Biometric Recognit.*, Sep. 2016, Springer, pp. 68–76. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-46654-5_8#citeas

[8] H. Fan and E. Zhou, “Approaching human level facial landmark localization by deep learning,” *Image Vis. Comput.*, vol. 47, pp. 27–35, Mar. 2016.

[9] W. W. Greulich and S. I. Pyle, “Radiographic atlas of skeletal development of the hand and wrist,” *Amer. J. Med. Sci.*, vol. 238, no. 3, p. 393, Sep. 1959.

[10] G. E. Güraksin, H. uğuz, and Ö. K. BAYKAN, “Bone age determination in young children (newborn to 6 years old) using support vector machines,” *Turkish J. Electr. Eng. Comput. Sci.*, vol. 24, no. 3, pp. 1693–1708, Mar. 2016.

[11] S. Honari, P. Molchanov, S. Tyree, P. Vincent, C. Pal, and J. Kautz, “Improving landmark localization with semi-supervised learning,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1546–1555.

[12] M. Jaderberg, K. Simonyan, and A. Zisserman, “Spatial transformer networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.

[13] Y. Jia *et al.* (2014). “Caffe: Convolutional architecture for fast feature embedding.” [Online]. Available: https://arxiv.org/abs/1408.5093

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[15] D. B. Larson *et al.*, “Performance of deep-learning neural network model in assessingskeletal maturity on pediatric hand radiographs,” *Radiology*, vol. 287, no. 1, Nov. 2017, Art. no. 170236.

[16] D. G. Lowe, “Object recognition from local scale-invariant features,” in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 1150–1157.

[17] M. Maravic, C. Berge, J. Daures, and M. Boissier, “Practices for managing a flare of long-standing rheumatoid arthritis: Survey among French rheumatologists,” *Clin Exp. Rheumatol.*, vol. 23, no. 1, pp. 36–42, Jan. 2005.

[18] M.Á. Martín-Fernández, R. Cárdenes, E. Muñoz-Moreno, R. de Luis-García, M. Martín-Fernández, and C. Alberola-López, “Automatic articulated registration of hand radiographs,” *Image Vis. Comput.*, vol. 27, no. 8, pp. 1207–1222, Jul. 2009.

[19] R. P. Mihail, G. Blomquist, and N. Jacobs, “A CRF approach to fitting a generalized hand skeleton model,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2014, pp. 409–416.

[20] A. Pfeil *et al.*, “Joint damage in rheumatoid arthritis: Assessment of a new scoring method,” *Arthritis Res. Therapy*, vol. 15, no. 1, p. R27, Feb. 2013.

[21] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, 2015, Springer, pp. 234–241.

[22] J. A. Singh *et al.*, “2015 American college of rheumatology guideline for the treatment of rheumatoid arthritis,” *Arthritis Rheumatology*, vol. 68, no. 1, pp. 1–26, Jan. 2016.

[23] C. Szegedy *et al.*, “Going deeper with convolutions,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[24] Z. Tang, X. Peng, S. Geng, L. Wu, S. Zhang, and D. Metaxas. (2018). “Quantized densely connected u-nets for efficient landmarklocalization.” [Online]. Available: https://arxiv.org/abs/1808.02194

[25] V. Vezhnevets and V. Konouchine, “GrowCut: Interactive multi-label ND image segmentation by cellular automata,” in *proc. Graphicon*, vol. 1, Jun. 2005, pp. 150–156.

[26] L. Wang, X. Yu, and D. N. Metaxas, “A coupled encoder-decoder network for joint face detection and landmark localization,” in *Proc. 12th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Jun. 2017, pp. 251–257.