Classifying Treatment Effectiveness in Chronic Recurrent Multifocal Osteomyelitis from MRIs

Anna Merkoulovitch
Stanford University SCPD
annamerk@stanford.edu

Zach Wener-Fligner
Stanford University SCPD
zbwener@stanford.edu

***Abstract*—Chronic Recurrent Multifocal Osteomyelitis (CRMO) is a rare condition mainly affecting the distal regions of long bones in the body. We present a proof of concept application of machine learning to predict disease progression on pairs of MRI images containing the knee and long bones of the leg. In this approach, we train multiple classifiers: logistic and kNN classifiers with features extracted using a pre-trained Inception-v3 CNN and SVM and Naive Bayes classifiers on a *bag of visual words.* We use ensemble voting to combine these models and present results for both multi-class (*improved*; *persisted*; and *regressed*) and binary classes (*improved*; and *persisted*/*regressed*).**

Keywords—CRMO, MRI, machine learning, ensembling, bag of visual words, inception v3, convolutional neural network

I. Introduction

Chronic Recurrent Multifocal Osteomyelitis (CRMO) is an inflammatory bone disease, affecting primarily children, where patients present with bone pain and localized swelling. Since its first description in 1972, there have been over 500 documented cases. Clinically, CRMO affects mainly the distal regions of long bones including the femur and tibula [1].

During CRMO treatment, physicians use whole-body magnetic resonance imaging (WB-MRI) to evaluate patient disease progression and response to treatment [1]. The primary indicators of disease progression are the presence of lesions and localized brightness in the affected regions, with decreases in both corresponding to an improvement in condition.

Here, we evaluate disease progression automatically using machine learning on pairs of images from MRI scans. Due to the complexity of analyzing and generating high-quality features for a full body MRI, we consider only subsets of an MRI that contain clear images of the knee and long bones of the leg. This simplifies the problem and enables classical machine learning techniques. Pairs of input MRI images are classified as members of three classes: *condition improved*; c*ondition persisted*; and *condition regressed.* Additional models for the binary class problem: *condition improved*; and c*ondition persisted or regressed* are also presented.

Convolutional neural networks (CNNs) that are used for many computer vision and MRI processing applications require large datasets to avoid overfitting on the training set. Such requirements are unreasonable in many clinical settings, particularly in rare diseases such as CRMO where the number of known cases are in the hundreds. Instead, we focus on careful data augmentation and methods such as cross-validation and regularization to reduce overfitting. We use custom methods to extract features from images, applying classical models to attempt to make global predictions about an individual's disease progression based on an MRI subset.

II. Related Work

A. Chronic Recurrent Multifocal Osteomyelitis Research

WB-MRIs have proven to be a valuable tool in clinical and research CRMO settings. Roderick et al. evaluated CRMO treatment effectiveness by considering changes in visible lesions in WB-MRI [1]. In [2], Arnoldi et al. devised a radiologic index for non-bacterial osteitis (RINBO), including CRMO, that allows standardized reporting of WB-MRI. Still, the method involves manually counting the number of lesions and manual classification of lesion size. To our knowledge, no research has attempted fully-automated evaluation of CRMO.

B. Scene Change Detection

Evaluating pairs of MRI scans is similar to computer vision problems that attempt to recognize changes in scenes with applications to building and traffic monitoring [3]. Whereas traditional scene detection does gross object detection such as the addition of an object to a scene—ignoring changes to brightness—we are interested in detecting small changes to brightness and luminosity.

C. Machine Learning Applications to MRIs

Machine learning applied to MRI and other radiograph images is an active field of research [4,5,6,7]. Tiulpin et al. showed that CNNs trained on thousands of knee MRIs could be an effective classifier of osteoarthritis severity as measured by the Kellgren-Lawrence scale (in essence a more mature version of RINBO) [3]. Antony et al. applied CNNs to both knee joint detection and disease classification [7]. Non-neural network techniques have been studied as well; Wang et. al. applied a SURF variant for image registration in [8].

III. Data Preparation

WB-MRI scans from 45 patients were procured from the Bristol Royal Hospital, with scans averaging one year apart per patient. A dataset containing the original radiologists' assessment, based on clinical data and MRI readings, was also

procured. The radiologist assessments were simplified to fit a three-class model: given two consecutive MRI scans for a patient taken at two different dates, a patient's condition with respect to CRMO either *improved (I), regressed (R)*, or *persisted (S)*. As the scans represent a cohort of patients undergoing treatment with pamidronate therapy, the data is skewed towards the *improved* class.

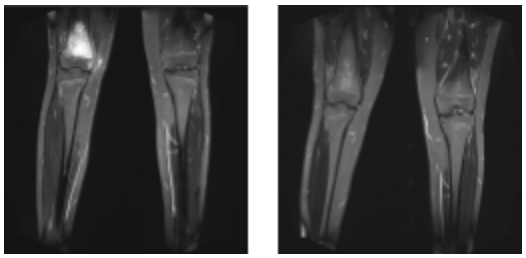

Fig. 1. Typical MRI appearances of improved CRMO condition after treatment with pamidronate therapy. (left: MRI dated 3-11-14, right: MRI dates 8-14-14)

The MRI dataset was provided in DICOM format, each scan consisting of approximately one thousand individual images depicting cross-sections of the body. From this initial data, a pared-down dataset was manually curated by selecting one to two representative images with clear views of the knee and long bones of the leg from each MRI scan. In some cases, patients had no high-quality representative images because all leg and knee images were extremely blurry or noisy; these scans were omitted from the set, leaving scans of 28 patients.

A list of date pairs and disease progression labels was manually curated from the information provided by the radiologist. When possible, labels for non-consecutive scans were inferred to increase dataset size.

Merging the images with the curated radiologist data resulted in 55 examples, each consisting of a pair of images similar to figure 1, and a label of *I*, *S* or *R*. Image feature vectors were extracted in multiple ways: from a pre-trained convolutional neural network producing 2048-length vectors; and via a *bag of visual words* technique, producing vectors of length $2|V|$, where $|V|$ is the visual vocabulary size. These approaches are described in the *Methods* section.

A. Data Augmentation

Since initial models had high-variance issues, data augmentation was used to produce more training examples (Fig. 2). These were created by swapping the order of pairs of images such that the latter image was treated as the first scan and initial images were treated as the second scan, with requisite label adjustments. Additional image augmentation was performed for the *bag of visual words* model by random linear stretching and adding Gaussian noise, and is described in the *Experiments* section.

B. Data Distribution

The augmented dataset was split to create 56 training samples and 25 validation samples. A significant number of samples were placed in the validation set in an effort to realistically quantify the generalization error. An additional small set of 7 test examples was held-out as a means for assessing final model quality at the end of development. The test sample was hand-selected prior to augmentation and contains a discrete set of patients whose scans do not overlap with those in the development sets.

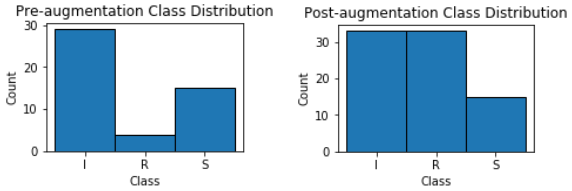

Fig. 2. Class distribution of train/dev sets before and after augmentation.

IV. Methods

We present two component approaches and an ensembling technique producing the prediction pipeline shown in figure 3.

A. Transfer Learning with Inception-v3 network

Transfer learning is the process of using models trained in one setting for application to other problems. The Inception-v3 model presented by Szegedy, C., et al [9], trained for the ImageNet image classification challenge, achieved a top-1 error rate of 4.2% and has been applied to a variety of machine learning applications since.

We used the convolutional base of the pretrained model and extracted features generated in the penultimate layer using Tensorflow [10, 11]. The final layer was then replaced by a custom predictor, allowing us to build convolutional models in a small dataset setting where retraining a CNN is not possible.

Images were first normalized to match the training size of Inception-v3 (299x299) by padding and resizing, before being run through the network. Multiple approaches were tried to convert individual image feature vectors into a format to represent an MRI pair. Predictor models were trained using two classes of models:

a) ***Softmax Regression***: a generalization of logistic regression that applies to multi-class problems by defining linear boundaries between classes. Softmax uses the multi-class cross-entropy loss function (4.1) with probability determined by the softmax function (4.2):

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{K=1}^{3} y_K^{(i)} \log h_\theta(x^{(i)}) + \lambda||\theta||_2^2 \tag{4.1}$$

$$h_\theta(x^{(i)}) = \frac{e^{\theta_{y_K}^T x^{(i)}}}{\sum_{j=1}^{K} e^{\theta_j^T x^{(i)}}} \tag{4.2}$$

where m is the number of examples, K is the class and $y^{(i)}$ is the one-hot class encoding for i. The regression term $\lambda||\theta||_2^2$ limits overfitting by penalizing large parameters to improve generalizability. This is important in small datasets where overfitting is often a problem.

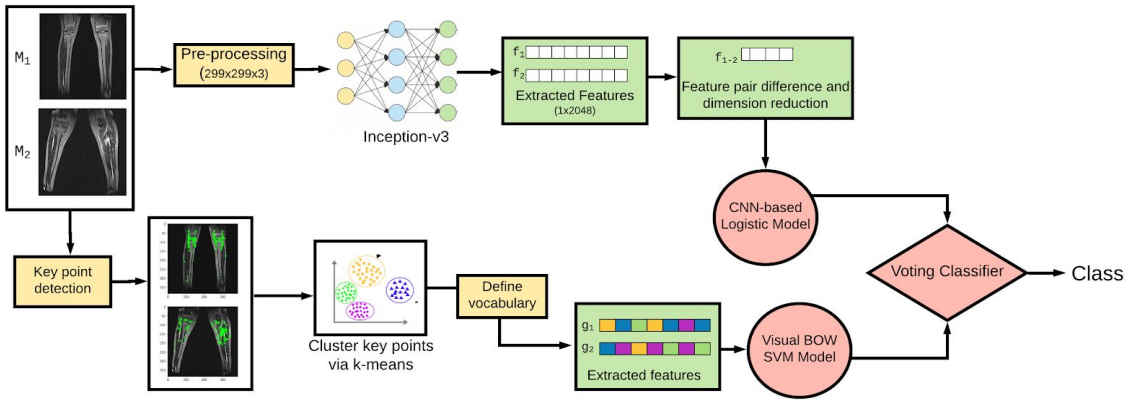

Fig. 3. Overall architecture for proposed model.

b) ***K-Nearest Neighbors*****:** an item is classified by majority voting of k neighbors.

B. Bag of Visual Words

The *Bag of Visual Words* [12] is a technique for transforming a dataset of images into a feature set, where a given image is represented as a set of common visual components, termed *visual words*. First, a *visual vocabulary* $V = \{\mu_1, \ldots, \mu_n\}$ is assembled by extracting a set of *key points* for each image in our dataset, and running a k-means clustering algorithm on the set of key points. Let the set of key points for image i be F_i. Then the set of all key points across all images is represented in (4.3) with corresponding k-means clusters given by (4.4):

$$X_{kp} = \bigcup_{i=1}^{m} F_i \tag{4.3}$$

$$c^{(i)} := \arg\min_j ||x^{(i)} = \mu_j||^2; \mu_j := \frac{\sum_{i=1}^{m} \mathbb{I}\{c^{(i)} = j\} x^{(i)}}{\sum_{i=1}^{m} \mathbb{I}\{c^{(i)} = j\}} \tag{4.4}$$

for i from 1 to m and j from 1 to n. Then, a single image can be represented by a vector in $\mathbb{R}^n$, where the ith element is the number of times a feature whose closest centroid is μ_i appears in the image. The input to our model is represented in $\mathbb{R}^{2n}$, acquired by concatenating vectors for two input images.

On the resulting dataset, we trained SVM with radial basis function kernel and Naive Bayes classifiers. For Naive Bayes, a Bernoulli event model was used, and the feature vectors were transformed to a binary format, with 1 indicating the presence of a centroid in the image and 0 indicating its absence. Then, predictions are given by (4.5) and (4.6):

$$\hat{y} = \arg\max_c \frac{\prod_{i=1}^{n} p(x_i \mid y_c) p(y_c)}{\sum_{c'} \prod_{i=1}^{n} p(x_i \mid y_{c'}) p(y_{c'})} \tag{4.5}$$

$$p(x_i \mid y_c) = \frac{\sum_{j=1}^{m} \mathbb{I}\{y^{(j)} = c \wedge x_i^{(j)} = 1\}}{\sum_{j=1}^{m} \mathbb{I}\{y^{(j)} = c\}} \tag{4.6}$$

X_{kp} was generated using Scale-Invariant Feature Transform (SIFT)[13]; Speeded-Up Robust Features (SURF) [14]; and Oriented FAST and Rotated BRIEF (ORB) [15].

Both SURF and SIFT rely on approximations for the Laplacian of Gaussians (LoG), an edge detection technique that applies a Laplacian operator to a Gaussian-blurred image. The Gaussian blur is a low-pass filter, reducing noise, and the zeros of the Laplacian filter—a second-order differential operator—indicate areas of rapid change, characterizations of edges and blobs. SIFT approximates the LoG with a Difference of Gaussians (DoG), a similar technique which takes the difference of two images blurred by Gaussians with different σ^2. SURF does the approximation using box filters, where a pixel is set to the average of its neighbors. ORB detects key points via a corner detector that uses heuristics on points within a given radius of a pixel.

After dropping low-contrast points and edge points, the remaining key points are transformed to vector descriptions computed by examining a window around the point.

C. Voting Ensemble

Ensembling methods were used to decrease variance by combining predictions of multiple weaker classifiers. We used a simple voting ensemble to combine the individually trained models discussed above. Soft weighting was used, which considers the probabilities produced by component models, whereas hard voting considers only the predicted class.

V. Experiments and Results

Component models and parameter tuning were performed on a 70:30 train/dev split. We used 5 to 7-fold cross validation, ensuring sets were large enough to represent all classes in each round. The best-performing models were re-trained on the full training and development sets before final evaluation on the held-out test set. As accuracy is a poor lone indicator of success in small dataset problems, we use several metrics:

F-1 Score. Weighted average that attempts to balance false positives and negatives by considering precision and recall.

Receiver Operator Curve (ROC). Plots recall against (1 - specificity); is a visualization of the discriminatory power of a model versus random guessing (a straight line). AU-ROC, the

area under the curve, summarizes the ROC in a scalar between 0 and 1. AU-ROC closer to 1 indicates higher quality.

Macro/Micro Averaging. Used in multi-classification problems; macro average treats all classes equally, while micro average weights classes based on class size.

A. Baseline

We set a baseline by subtracting image brightness histograms, creating length-256 vectors, and training logistic regression. This performed similar to random guessing.

B. Transfer Learning Model.

In the transfer learning approach, we focused on methods for transforming individual feature vectors generated by the CNN into features that performed successfully when representing a pair of images. Two primary methods for combining data were tested: *feature concatenation,* where individual feature vectors were concatenated to form vectors of length 4096; and *feature dissimilarity*, where we subtract the individual feature vectors, giving a vector of length 2048.

Experiments were also conducted to process the initial data by performing data augmentation, feature normalization and feature reduction (removing features with $\sigma^2 < 0.1$). Table I shows the evaluation set F-1 and accuracy scores after being trained with default sklearn logistic regression parameters. In all instances, micro average gives a more favorable result, likely due to a shortage of samples in the *persisted (S)* class, which is also the primarily misclassified class. The bolded row represents the processing method that best balanced scores.

Feature dissimilarity, applied to video scene detection in [9], outperformed simple concatenation. Unfortunately, since CNN features are not interpretable, there is ambiguity in what the dissimilarity represents. Data augmentation and feature reduction unsurprisingly improved model performance, given the small dataset's proneness to overfitting and high variance.

TABLE I. FEATURE SET SELECTION OVER MULTIPLE PROCESSING STRATEGIES

Data Adjustment			*F-1 Score*		*Accuracy*
Aug	***Norm***	***FR***	***Micro***	***Macro***	
Feature Concatenation					
			0.80	0.52	0.8
			0.40	0.44	0.44
			0.56	0.54	0.56
Feature Dissimilarity					
			0.67	0.41	0.66
			0.73	0.45	0.73
			0.72	0.62	0.72
			0.76	**0.66**	**0.76**

*Aug: Data Augmentation. Norm: feature normalization, FR: feature reduction
*Gray boxes represent adjustment made on data

Hyperparameter tuning was performed via grid search, selecting for models with high F-1 scores and better generalization error. Both multi-class and binary models performed best using L2 regularization with λ=10 and λ=20 respectively, with performance shown in Table II.

TABLE II. F-1 SCORES OF TRANSFER LEARNING MODELS

Model	***Train***	***Dev***	***Test***
Multi-class	0.92	0.79	0.36
Binary classification	0.95	0.88	0.71

Multiclass values represent macro-averaged f1-score

The confusion matrix in Figure 4 indicates that softmax regression had difficulty distinguishing the *persisted (S)* class in the multi-class problem. This suggests this class may not be linearly separable, which is interesting since it class sits between the others in terms of disease progression. Interestingly, in the binary problem, once *S* was grouped with *R*, the model was quite successful with an AUC of 0.92.

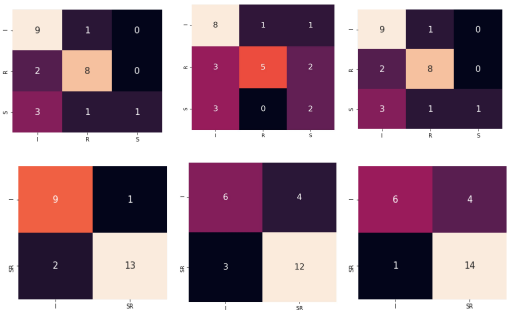

Fig. 4. Confusion matrices for best performing models. (Top (multi-class): softmax; visual words; ensembled. Bottom (binary): logistic; visual words; ensembled.

C. Visual Bag of Words

We trained Naive Bayes and SVM with rbf kernel classifiers on the visual word datasets with vocabularies ranging from size 5 to 500. Hyperparameters for the SVM were selected via grid search in 5-fold cross validation. The highest-performing hyperparameters for the SVM models were found at C=10, $\gamma = 0.0001$. In Naive Bayes we experimented with both a uniform prior for visual world probabilities and with priors learned from the training set. The best-performing models during cross-validation are summarized in Table III.

TABLE III. VISUAL BAG OF WORDS PERFORMANCE (F-1 SCORING)

Model	***Train***	***Dev***	***Test***
Multi-class	1.0	0.67	0.22
Binary classification	1.0	0.37	0.17

Multiclass model: Naive Bayes, SIFT, |V|=500. Binary model: SVM, ORB, |V|=50.
Multiclass values represent macro-averaged f1-scores

Both bag-of-words models were high-variance, achieving 100% per accuracy on the training set. We augmented the dataset by generating 10-20 synthetic images for each training image, created by applying a random linear warp and adding Gaussian noise. This prevented the model from achieving perfect training accuracy, but failed to improve cross validation performance.

This method consistently struggled to generalize past the training set. Despite key point density around interesting regions in the image (visible as the green dots in the

architecture diagram of fig 3), it is possible that the key points are not effective proxies for the indicative regions in CRMO or that our dataset was too small to extract useful visual word representations. In addition, the scale-invariant properties of the feature extractors may make them ill-suited to transformation and noise-based data augmentation.

D. *Voting Ensemble*

Using the hypertuned parameters for models in previous sections, we built custom pipelines to extract relevant features for each model and combine resulting predictions. The same cross-validation grid search approach was used to determine optimal weighting of the component models. Perhaps unsurprisingly, the highest performing ensemble models gave higher weight to the CNN-based model (e.g. 3:1 for multiclass). Although the confusion matrices on the validation set look the same for transfer learning and the ensembled method (see Figure 4), comparing F-1 scores for the corresponding cross-validation sets (see Tables II, IV) suggests ensembling did help reduce variance.

TABLE IV. ENSEMBLED MODEL PERFORMANCE (F-1 SCORING)

Model	***Train***	***Dev***	***Test***
Multi-class	0.95	0.63	0.42
Binary classification	0.89	0.78	0.71

Multiclass values represent macro-averaged f1-scores

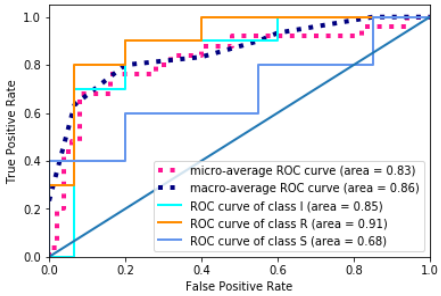

Fig. 5. Receiver Operator Curve for multi-class ensembled model.

The ensembled models predicted 4/7 examples correctly in the multiclass problem, and 5/7 correctly in the binary problem. Interestingly, visual inspection of the misclassifications showed that two of the examples in the multiclass and one of the examples in the binary class corresponded to images from a patient labeled *S*, that clearly matched our intuitive understanding of the *I* class, suggesting possible human error. Still, some of the other misclassified examples were harder to interpret without CRMO expertise.

More generally, our simplified model of isolating the knees of patients inherently adds label uncertainty. Although CRMO is most commonly found in the legs, previous analysis on this cohort showed lesions throughout the body [1], and so the hand-selected images may not be representative of the actual labels generated from the full WB-MRI.

Consistently, all multi-class models were unable to properly predict class *S* (see Figure 4). This may be due to a smaller number of samples in the class, or to difficulty extracting features representative of this class compared to *I* or *R* classes. Poor predictability power of *S* can also be seen in the lower AU-ROC value for the multi-class ensemble (Fig 5).

Although we made efforts to address model variance by ensembling, decreasing feature size, and performing data augmentation, the majority of final models still struggled with high variance. A validation curve for varied values of regularization parameters (Figure 6) shows that decreasing regularization makes our model swing from high bias to high bias and variance, suggesting more focus is still needed on improving features and increasing dataset size.

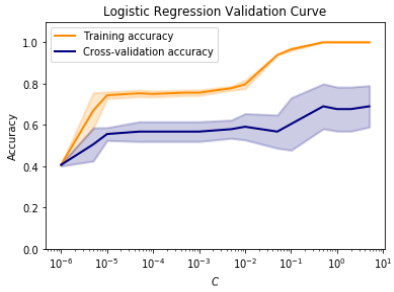

Fig. 6. Validation curve of training and cross-validation accuracy for varied C=1/λ parameter values in logistic regression.

VI. CONCLUSION AND FUTURE WORK

Our results are a promising beginning to research applying machine learning to CRMO, and we believe further investment in the small-data techniques is worthwhile. The most promising future work in improving our model lies in improving our features, as experimenting with models and hyperparameters had relatively little impact on our results compared to feature selection. The CNN transfer learning approach showed a level of success, and we suspect it could be improved by retraining on the Inception-v3 network using a more relevant MRI dataset (rather than the original ImageNet set). Such data exists for conditions like osteoarthritis and might be applied here to extract more appropriate features.

Working closely with CRMO experts is another promising path forward. Features based on RINBO developed with expert input could be information-dense, keeping both bias and variance low. Interpretable models such as decision trees could also make our models more useful in a clinical setting.

ACKNOWLEDGMENT

We thank Dr. Athimalaipet V. Ramanan of Bristol Royal Hospital for introducing us to CRMO research and guiding us along the way; Dr. Chandrika S. Bhat for assisting in manually curating data collected from radiologists; and the radiology staff for anonymizing and sharing patient MRI images.

We also thank TA Fantine Huot for her tremendous amount of guidance and support throughout the project, and TA Suvadip Paul for assistance in connecting with the UK-based team and technical guidance.

References

[1] Roderick et al. (2016). Chronic recurrent multifocal osteomyelitis (CRMO) - advancing the diagnosis. *Pediatric Rheumatology.*, 14:47. doi: 10.1186/s12969-011-0109-1

[2] Arnoldi A.P. et al. (2017). Whole-body MRI in patients with non-bacterial Osteitis: Radiological findings and correlation with clinical data. *Eur Radiol.* 27:2391-299. doi:10.1007/s00330-016-4586-x

[3] Sakurada, K., & Okatani, T. (2015, September). Change Detection from a Street Image Pair using CNN Features and Superpixel Segmentation. In *BMVC* (pp. 61-1).

[4] Lavdas, I., et al. (2017). Fully automatic, multi-organ segmentation in normal whole body magnetic resonance imaging (MRI), using classification forests (CFs), convolutional neural networks (CNNs), and a multi-atlas (MA) approach. *Med. Phys.*, 44: 5210-5220. doi:10.1002/mp.12492

[5] Dazhou Guo D., et al. (2015). Automated lesion detection on MRI scans using combined unsupervised and supervised methods. *BMC Med Imaging.* 15: 50. Published online 2015 Oct 30.

[6] Tiulpin, Aleksei, et al. "Automatic knee osteoarthritis diagnosis from plain radiographs: a deep learning-based approach." *Scientific reports* 8.1 (2018): 1727.

[7] Antony, Joseph, et al. "Automatic Detection of Knee Joints and Quantification of Knee Osteoarthritis Severity using Convolutional Neural Networks." *International Conference on Machine Learning and Data Mining in Pattern Recognition*. Springer, Cham, 2017.

[8] Wang, Anna, et al. "Research on a novel non-rigid registration for medical image based on SURF and APSO." *Image and Signal Processing (CISP), 2010 3rd International Congress on*. Vol. 6. IEEE, 2010.

[9] Szegedy, C., et al. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).

[10] Abadi, Martín, et al. "Tensorflow: a system for large-scale machine learning." *OSDI*. Vol. 16. 2016.

[11] Image classification with a pre-trained neural network. (2016, June 21). Retrieved from: https://www.kernix.com/blog/image-classification-with-a-pre-trained-deep-neural-network_p11

[12] Sivic, J., & Zisserman, A. (2006). Video Google: Efficient visual search of videos. In *Toward category-level object recognition* (pp. 127-144). Springer, Berlin, Heidelberg.

[13] Lowe, David G. "Distinctive image features from scale-invariant keypoints." *International journal of computer vision.* 60.2 (2004): 91-110.

[14] Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. "Surf: Speeded up robust features." *European conference on computer vision*. Springer, Berlin, Heidelberg, 2006.

[15] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, November). ORB: An efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE international conference on* (pp. 2564-2571). IEEE.

[16] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.

[17] Bradski, Gary, and Adrian Kaehler. "OpenCV." *Dr. Dobb's journal of software tools* 3 (2000).

[18] Kluyver, Thomas, et al. "Jupyter Notebooks-a publishing format for reproducible computational workflows." *ELPUB*. 2016.

Contributions

Anna Merkoulovitch

- Worked with pediatricians at the Bristol Children's Hospital to acquire MRI images and labels indicating whether a patient's condition "improved", "stayed the same", or "regressed".
- Project scoping strategy
- Visual image selection
- Manual data curation
- Baseline feature extraction, dataframe preparation, producing initial model
- Project proposal, poster
- Report writing & editing
- Shared code to perform GridCV parameter fitting, and creating confusion matrices
- Inception-v3 based model development
- Vote ensembling model development

Zach Wener-Fligner

- Worked with pediatricians at the Bristol Children's Hospital to acquire MRI images and labels indicating whether a patient's condition "improved", "stayed the same", or "regressed".
- Manual DICOM extraction from Horos
- Programmatic extraction of radiologist-labeled disease instances
- Visual image selection
- Manual data curation
- Project proposal, poster
- Report writing & editing
- Binary image thresholding investigation
- Script for comparing incorrectly-predicted images
- Bag of visual words model development

Link to code: https://github.com/annamerk/crmo-diagnosis-using-mri/