Skin Cancer Detection Using Sparse Coding

Tomás Miguel Donga Cardoso
tomas.cardoso@ist.utl.pt

Instituto Superior Técnico, Lisboa, Portugal

June 2019

Abstract

Melanoma of the skin is one of the deadliest cancer types. As is for most types of cancer, its chances of being cured greatly increase with the swiftness of its diagnosis. Due to this, great efforts have been put into using machine learning to automate the process of melanoma detection. This thesis joins that field of work by making use of sparse coding techniques embedded in a complete system for melanoma diagnosis that incorporates feature extraction and classification.

The methods used in this thesis are based on the sparse representation of data and dictionaries learned from data. The effectiveness of discriminative dictionaries and sparse codes is studied, as well as the application of hierarchical clustering to the dictionary atoms in order to further cut redundancies. Finally, the impact of deep learning in the aforementioned system is inspected through the use of deep features extracted from a pre-trained *convolutional neural network* (CNN), namely the VGG19 [28]. The systems proposed in this thesis achieve a sensitivity of $56,41\%$ and a specificity of $71,43\%$ for the image dataset from 2017 challenge from the International Skin Imaging Collaboration (ISIC) and a sensitivity of $64,84\%$ and a specificity of $88,82\%$ for the image dataset from the Interactive Atlas of Dermoscopy (EDRA).

Keywords: melanoma detection, sparse coding, svm, convolutional neural networks, hierarchical clustering, discriminative learning

Introduction

Cancer is one of the most deadly diseases that that currently afflicts the human kind. Amongst all types of cancer, skin cancer is the most common one and melanoma is the most aggressive and deadly type of skin cancer [1]. For melanoma, as is with most cancers, an early diagnosis is pivotal in order to reduce the mortality rate of those afflicted by it. Due to this, and with the uprise of machine learning and deep learning methods, increasing effort has been put into place in order to apply these methods to automate and facilitate the process of skin cancer detection [17] [23].

There are medical procedures that are used by medical experts to diagnose skin lesions, the most prominent ones being pattern analysis [30], ABCD rule [32] and seven-point checklist [5]. What all these have in common is that they focus on the analysis of dermoscopic features in the lesion, also called dermoscopic criteria. The presence of these features varies for every type of skin lesion, and even within the same type, every lesion is unique, which is why it is sometimes so difficult to correctly diagnose them. Hence, CAD systems trained from thousands of images may play an important role in assisting medical doctors in skin cancer detection. Which is why the goal of this thesis is focused on the development of such computer-aided diagnostic systems.

For the past three decades, different approaches were proposed to tackle the problem of skin cancer detection. The first aproaches were systems that took decisions based on global hand-crafted features such as color features [10], [29], texture features [9], [14], [25], border features [12], [15], [4], [22] and asymmetry features, including shape symmetry [22], [31], [27] and color and structure symmetry [22], [36]. Classifiers trained on dictionary-based features such as bag-of-words or sparse coding [7], [25], [26] were rarely used. Recently, deep learning started to be employed and is gradually becoming the standard in the area, making use of methods such as deep neural networks and transfer learning to achieve state-of-the-art results [20], [21], [24].

The goal of this thesis is to study the effectiveness of sparse coding techniques applied to the problem of melanoma skin cancer detection. This will be done through a number of experiments with different methods and algorithms in order to achieve the best possible result while also comparing it to a common baseline system.

This extended abstract will thus be organized as

follows:

- Section 1: Introduction
- Section 2: Sparse Representation
- Section 3: Baseline System
- Section 4: Discriminative Dictionary Learning
- Section 5: Deep Features
- Section 6: Experimental Results
- Section 7: Comparison and Assessment of the Proposed System
- Section 8: Conclusion

Sparse representation

Sparse coding

The goal of sparse representations is to replicate the input as closely as possible through a linear combination of atoms from a dictionary $\mathbf{D}$, while enforcing the sparsity of said linear combination. The vector of coefficients of the linear combination is called the sparse code and the sparsity of a vector increases as the number of non-zero elements in the vector decreases.

How close the sparse representation is to the input can be expressed as $||\mathbf{x}-\mathbf{D}\boldsymbol{\alpha}||_2^2$ which measures the reconstruction error, where $||\cdot||_2^2$ represents the square of the Euclidean norm (l_2 norm), $\mathbf{x}$ is the input, $\mathbf{D}$ the dictionary and $\boldsymbol{\alpha}$ the sparse codes.

The best way to enforce sparsity would be to ensure that the "l_0 norm" of the sparse code is low, since said norm counts the number of non-zero elements in a vector. But since it is a cardinal function, it is non-differentiable and difficult to optimize over. The "l_1 norm" is often used in its place, defined as

$$||\boldsymbol{\alpha}||_1 = \sum_{j=1}^{k} |\alpha^j|. \tag{1}$$

The l_1 norm sums over the absolute value of the elements of a vector and as such, a low norm value then indicates a sparser vector. The l_1 norm also makes the problem of obtaining α convex and with a unique solution, which facilitates the computation [33]. So, given an input $\mathbf{x}$, and a dictionary $\mathbf{D}$, the optimization problem to compute the corresponding sparse code is formulated as

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}||\mathbf{x}-\mathbf{D}\alpha||_2^2 + \lambda||\boldsymbol{\alpha}||_1. \tag{2}$$

where, for every training sample there is a minimization over the sparse code $\boldsymbol{\alpha_i}$ which involves a trade-off between minimizing the reconstructing error and enforcing the sparsity of the sparse code. This trade-off between the two terms is controlled by the variable λ.

Dictionary learning

However, the dictionary is often not known and therefore needs to be estimated from a training set of data. This is done by encapsulating the optimization over the sparse codes (2) with another optimization over the dictionary, that will use the input data and the estimations of sparse codes to learn a dictionary. Assuming that instead of a single input vector $\mathbf{x}$, there is a set of input vectors $\mathbf{X} = [\mathbf{x_1}, ..., \mathbf{x_p}]$, dictionary learning involves solving the optimization problem

$$\min_{\mathbf{D}} \frac{1}{p}\sum_{i=1}^{p} \min_{\boldsymbol{\alpha_i}} \frac{1}{2}||\mathbf{x_i}-\mathbf{D}\boldsymbol{\alpha_i}||_2^2 + \lambda||\boldsymbol{\alpha_i}||_1. \tag{3}$$

This optimization problem translates to solving two alternate, iterative optimization problems, one over the dictionary and the other over the sparse codes, until convergence.

Baseline System

In this section an initial baseline system for melanoma detection is proposed and described. The problem addressed is a binary classification problem: given a dermoscopic image of a skin lesion, the system should be able to distinguish between melanoma and benign skin lesions.

The baseline system will serve as a stepping stone for improvement and comparison troughout this thesis. The architecture of the baseline system is shown in figure 1 and was inspired by the system used in [8].

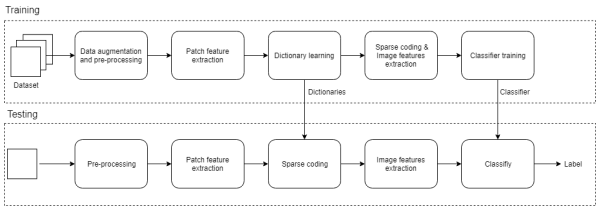

Figure 1: Block diagram of baseline system

The CAD system involves two modes (training and testing modes), each of them comprising several tasks.

Data augmentation and pre-processing

Both datasets are unbalanced class-wise, melanoma images account for only a small part of the dataset. The small presence of one of the classes hampers the training of the classifier which then produced subpar results. To tackle this problem, data augmentation techniques were used. For every melanoma image in the training set, three additional images were generated by rotating the image with multiples of 90.

Each image is also pre-processed. The size of the skin lesion and the percentage of the image it occupies varies greatly among the dataset, which means some images show a lot of the neighboring skin while others do not. A segmentation step is performed that crops a bounding box around the lesion, in order to focus the feature extraction procedure on the lesion, discarding the healthy skin, thus emphasizing lesion features. It is also important however not to crop immediately at the border of the lesion, since transition from healthy skin to lesion convey useful information about malignancy of the lesion. Finally, all images have varying aspect ratios, it is therefore important to maintain these when cropping as to not distort the images.

The images in the datasets are captured using different dermatoscopes and under different conditions, which results in images with different color spectra. This introduces significant color differences across images, which makes the classification task more difficult since similarities between in-class images are harder to find. The images are therefore passed through a *color normalization* function [6].

Patch feature extraction

Once a given image is loaded and goes through preprocessing, its features are extracted. In order to do this, the image is broken into non overlapping patches of size 16×16 pixels and local features are extracted from each of these. Two types of features were used: color and texture features, which are the two main sources of information for a dermoscopic image.

The chosen color features are color histograms. Each pixel in the patch has three color components (RGB channels). The color content of each patch is characterized by three color histograms. The histograms hold the information for the distribution of the three color channels in that specific patch, for that color channel. So, for every patch P_c, $c \in \{1, 2, 3\}$, 3 histograms h_c with $B = 16$ bins each are calculated as

$$h_c(i) = \frac{1}{L^2} \sum_{x=1}^{L} \sum_{y=1}^{L} b_i(P_c(x, y)), \quad i = 1, ..., B, \quad (4)$$

where $L \times L$ is the size of patch P_c, and $b_i(P_c(x, y))$ is the characteristic function of the ith bin of histogram h_c, that is 1 if the pixel $P_c(x, y)$ belongs to the ith histogram and 0 otherwise.

For the texture features, gradient histograms were used. The gradient of an image conveys information on the intensity changes near each pixel. In this case, the gradient of the gray-scale image is computed for every patch. At every pixel (x, y), the horizontal and vertical components of the gradient, $g_1(x, y)$ and $g_2(x, y)$, are computed using Sobel masks. These two components are then used to calculate the gradient magnitude $||g(x, y)||$

$$||g(x, y)|| = \sqrt{g_1^2(x, y) + g_2^2(x, y)}. \quad (5)$$

The gradient magnitude information are then used to build yet another texture histogram, for every patch, in a similar way to what is done with the color histograms:

$$h_m(i) = \frac{1}{N} \sum_{x=1}^{L} \sum_{y=1}^{L} b_i(||g(x, y)||), \quad i = 1, ..., B_m. \quad (6)$$

It should be noted that as texture feature, not only gradient magnitude was used, but also gradient orientation was experimented with. However, this type of feature achieved subpar results when comparing to color histograms and gradient magnitude histograms, therefore it was discarded.

Dictionary learning and sparse coding

In this problem we have two different sets of features (color histograms and gradient magnitude), that extract different types of information about the image. Due to this, two dictionaries were learned, one for each type of features. The input for dictionary learning are the local feature histograms obtained before, from all the training patches.

Image features extraction

Image features, as the name indicates, are global features that characterize the image as a whole. Therefore, the available sparse codes for a given image, one per patch, must be transformed into a single vector that describes the image.

A sparse code, by definition is a sparse vector of weights that multiplies the atoms in the dictionary to approximate the patch features. If looked at in a different perspective, a sparse code gives information on which atoms are important to characterize its respective patch features. In the context of this problem, the aggregation of sparse codes for a given image gives the importance of each atom in representing the whole image. These weights are then the chosen features to represent the image. However, since the images have different sizes, the number of sparse codes also varies from image to image which means that they cannot be directly used as features. To circumvent this, a histogram is made averaging the sparse codes of a given image

$$h_s = \frac{1}{p} \sum_{i=1}^{p} |\alpha_i|, \quad (7)$$

which in turns gives the average use of every atom in representing the patches of that image.

Classifier
The chosen classifier for the baseline system is the *Support Vector Machine* (SVM) [11]. The SVM is a popular classifier that has been applied to a wide range of problems. In its simplest version, it tries to learn an hyperplane that separates the training samples from two classes. The hyperplane can be replaced by more complex surfaces by using kernel functions, like the RBF kernel.

Two separate classifiers were trained, one for each type of features: color and gradient magnitude. The final predicted label for a given image comes from averaging the class-specific probabilities of that image given by both classifiers.

Experimental setup
With the standard parameters, the SVM classifier with the RBF kernel could not distinguish between melanoma and non-melanoma skin lesions as it would assign every sample to one of the classes. Further tuning was thus required.

Since the ISIC dataset supplies a separate test set, a k-folds cross-validation process is used to tune the hyperparameters of the system. This consists in splitting the training set into k parts (folds), one of which is used for validation and the others are used to train the classifier. The validation fold will rotate at each iteration such that all folds serve as validation fold.

A different procedure was adopted with the EDRA dataset as it does not have separate sets for training and testing and so, nested cross-validation was used. Nested cross-validation consists of two cycles, an inner cycle that performs normal cross-validation for model tuning and a outer cycle that evaluates the chosen model on the portion of the dataset that was not used in inner cycle of cross-validation. This allows for a computation of an average test score for the dataset.

The parameters that were tuned during the process for the SVM classifier were: *i)* the penalty parameter C for an error *ii)* the rbf kernel coefficient γ *iii)* the number of atoms in the dictionaries.

The results for both datasets using the baseline system are presented in table 1.

Discriminative Dictionary Learning

With the adoption of sparse representations in image classification tasks, it was not only necessary for the learned dictionaries to accurately represent the image, but also to provide discriminative information regarding the different classes. Therefore, different works started to propose strategies to enhance the discriminative properties of the dictionaries [16] [19] [35].

A discriminative dictionary contains specific subsets of atoms that are more specialized in a given class. This means that if the inputs show a high inter-class variability and low intra-class differences, they will tend to select the same atoms within the dictionary, as other inputs of the same class [16]. This enhances the ability to distinguish between inputs of different classes since their sparse codes would be very different, which is the goal of classification.

Concatenation of class-specific dictionaries
The simplest strategy to obtain a discriminative dictionary is to simply learn one dictionary using images from one class (melanoma training images) and learn another dictionary using images from the other class (non-melanoma training images), and then concatenate them.

To test the impact of using this type of discriminative dictionaries, the baseline was modified to include these dictionaries, replacing the ones discussed in section 3.3. The results for this new system for both datasets are presented in table 1.

Concatenation of class-specific Sparse Codes
A different approach for introducing discriminance between classes is here considered. Instead of concatenating the class-specific dictionaries and computing the sparse codes for the resulting dictionary, we will instead estimate two sparse codes for each image patch: one for the melanoma dictionary and another one for the non-melanoma. Then, we will concatenate them into one single sparse code. This is done for both types of features and two classifiers are trained, identically to the baseline. The results for this new system for both datasets are presented in table 1.

Clustering of dictionary atoms
A reason why discriminative dictionaries may not work well is the similarity between inter-class images, which results in the existence of atoms that are common to both classes in the class-specific dictionaries, which in turn results in a near even usage of atoms from both class-specific dictionaries by most images. A logical step to address this issue would be to remove the common atoms, in order to improve the discriminative properties of the dictionaries. This may constrain the images to use more class-specific atoms, improving the classification.

Ensuring that the class-specific dictionaries do not share common atoms has already been adopted in other works. In [16], the optimization problem for the dictionary learning is changed such that there are class-specific dictionaries that contain the most distinctive atoms which are used for classification, as well as a common dictionary, only used for representation.

A different strategy is adopted in this work. First, a separate dictionary is learned for each of the classes. Then, the two dictionaries are concatened.

Finally, similar atoms are removed using a hiearchical clustering algorithm. In the following subsection we detail the adopted clustering approach.

Hierarchical clustering

Given a collection of vectors, the goal of hiearchical clustering is to iteratively group them until there is only a single cluster, such that at the beginning each vector represents a cluster and by the end only one cluster will remain. At each step of the algorithm, two clusters are grouped together if they are the most "similar" ones. The "similarity" between clusters can be measured using a single or complete-linkage strategy. Different metrics can be used to compute the distances, such as euclidean distance, correlation, cosine distance, among others [34] [13].

In context of this thesis, in the first step of the algorithm each cluster corresponds to an atom in the dictionary. So, it is not desirable that only one cluster remains, that is, only one atom. Hierarchical clustering must be performed, until a stopping criterion is met. The criterion used in this thesis is that the maximal distance between clusters must be below a given threshold. This distance, defined as clustering threshold, is considered to be a hyperparameter of the model, which is tuned using cross-validation, similarly to the other hyperparameters. The atoms identified as common to both classes are placed in a separate dictionary and discarded. The remaining atoms from both class-specific dictionaries will be concatenated, as in section 4.1.

In the context of the system, this algorithm is then applied to both dictionaries. The resulting dictionaries are then used for sparse coding. The results for this new system with the inclusion of hierarchical clustering applied to dictionary atoms for both datasets are presented in table 1.

Deep features

Thus far, the focus on the improvement on the baseline system has been in the dictionary learning and sparse coding section, always using the hand-crafted features detailed in section 3. This chapter aims at investigating the use of a different kind of features, extracted from a convolutional neural network (CNN).

The chosen CNN is the VGG19 [28], a convolutional neural network with 19 layers, trained on the ImageNet database [2].

The VGG19 net is comprised of several blocks of convolutional layers, which will gradually reduce the size of the feature maps extracted by the convolution process. The features chosen to be extracted are from the fourth convolutional layer of the fifth block of layers. This choice was made taking in account their abstraction in relation to the original image, but mainly their size, which must not be too great due to hardware limitations. At the output of this layer are 512 activation maps of size 14×14, where each activation map will be a feature vector. This means that for every image, there are 512 feature vectors of size $14 \times 14 = 196$. These features will replace the hand-crafted ones on the previous systems.

One final consideration that must be made is that the VGG19 net only accepts square images of size 224×224, therefore all dermoscopic images had to be resized to fit this, this means that the aspect ratio of most images will be distorted, which may potentially hinder the results.

Deep features applied to discussed methods

The initial idea was to completely replace the hand-crafted features with these deep features, but we soon realized that the results it produced were no greater than the ones each individual classifier of the hand-crafted features did. Therefore, these deep features were looked at as a complement to the already existing hand-crafted features. A third dictionary and classifier are then trained using the deep features and the final label prediction is now pondered between the three classifiers, one for each type of features. This was done for each of the four systems proposed in this work, and the impact of deep features in the final result is seen in table 1.

Experimental Results

In this section are presented the results for all systems evaluated on both datasets: ISIC and EDRA.

Three metrics are chosen to evaluate each system. Sensitivity (SE) measures true positives. Within the context of this problem, it shows the ratio between correctly classified melanoma skin lesions and the total number of melanoma skin lesions in the test set. In a similar way, specificity (SP) measures the true negatives. Finally, the Balanced accuracy ($BACC$) is the average of sensitivity and specificity, and gives the average performance of the system on both classes. Taking this into consideration, the results are presented in table 1.

As can be seen in table 1, there is a large difference in the results between the EDRA and ISIC datasets, the latter being much worse. This is mainly due to the fact tat the ISIC dataset is quite difficult, specially its test set.

It is also observable that there it is not one system that achieves the best performance for both datasets. The system presented in section 4.3, with deep features included, achieves the best performance for the ISIC dataset with a $BACC$ of $63,50\%$, while the best performing system for the EDRA dataset is the one presented in section 4.3, with deep features excluded, achieving a $BACC$ of $76,83\%$.

Table 1: Performances on both datasets for all systems presented in this work.

Method	Fusion	ISIC	EDRA
3	2 classifiers	SE=47, 01 SP=75, 36 BACC=61, 19	SE=61, 40 SP=86, 51 BACC=73, 96
	3 classifiers	SE=47, 01 SP=75, 16 BACC=61, 08	SE=63, 37 SP=86, 95 BACC=75, 16
4.1	2 classifiers	SE=44, 44 SP=72, 88 BACC=58, 66	SE=64, 84 SP=88, 82 **BACC=76, 83**
	3 classifiers	SE=47, 01 SP=73, 09 BACC=60, 05	SE=62, 21 SP=88, 54 BACC=75, 38
4.2	2 classifiers	SE=41, 03 SP=75, 57 BACC=58, 30	SE=61, 24 SP=87, 59 BACC=74, 41
	3 classifiers	SE=47, 01 SP=76, 19 BACC=61, 60	SE=61, 29 SP=88, 82 BACC=75, 06
4.3	2 classifiers	SE=51, 28 SP=68, 74 BACC=60, 01	SE=61, 52 SP=83, 79 BACC=72, 66
	3 classifiers	SE=56, 41 SP=71, 43 **BACC=63, 50**	SE=61, 37 SP=86, 29 BACC=73, 83

Comparison and Assessment of the Proposed System

In this section, we compare our best performing system, presented in section 4.3 with the inclusion of deep features, with the participants of the 2017 ISIC challenge [3]. Some experiments to assess the the relevance of initialization in the dictionary estimation and the amount of data are also performed.

In the context of the ISIC 2017 challenge

The ISIC dataset is provided by the International Skin Imaging Collaboration, which holds a challenge every year on skin cancer detection. This particular dataset is from the 2017 edition [3],thus it is possible to compare the proposed model with the contestants of that edition.

It should first be noted that the participants for the sub-challenge of melanoma classification are ranked accordingly to the ROC AUC of their systems, not by the metrics used in this work and as such their systems are built to achieve the highest possible ROC AUC, in the same way ours is built to achieve the highest possible BACC. Therefore any comparisons between the ROC AUC and BACC of our system and the ones from the contestants should be made with reservations.

That being said, in the context of the ISIC 2017 competition leader board for the melanoma classification category, our system is on the $78,26\%$ percentile, which means that $78,26\%$ of the contestants achieved a higher ROC AUC than us, which is not very good. However, if the leader board was arranged according to the BACC, the proposed system would rank in the 39.13% percentile, which is significantly higher.

It should also be noted that the ISIC challenge allows for the use of external data, not provided by them. A fraction of contestants use external data and it may be unfair to compare their systems with those that are trained without extra data, given the relevant role training data plays in classifier performance. If the systems trained with extra data are then excluded, our system would be in the $71,43\%$ percentile for the ROC AUC and the $35,71\%$ for the balanced accuracy.

Relevance of Dictionary Initialization and Datasets

This section discusses the influence of dictionary initialization and of the training set in the final results.

Dictionary initialization

By inspection of the experimental results we realized that, not only the parameters (number of atoms, clustering threshold, C and γ for the SVM) tuned in the cross-validation influenced the performance on the test set, but the dictionary initialization as well. The dictionary initialization is handled

by the dictionary learning function from the spams package [18]. This toolbox randomly takes k feature vectors from the training set to serve as the initial atoms. This introduces a variability in the estimated dictionaries and influences the classification performance of the model.

We wanted to determine the degree of influence of the initialization process. Thus, we carried out a simple experiment that consisted of selecting the best configurarion of parameters for the ISIC 2017 obtained through cross-validation and the model described in section 4.3 with inclusion of deep features, learn a set of 100 dictionaries, and use them to train 100 classification systems. Then, the sensitivity, specificity, and balanced accuracy were computed for each of them, on the test set. The results are presented in figures 2, 3 and 4 respectively.

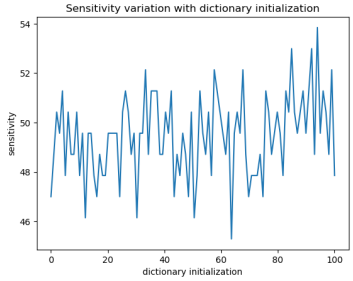

Figure 2: Variation of sensitivity, specificity and balanced accuracy with dictionary initialization for the ISIC dataset.

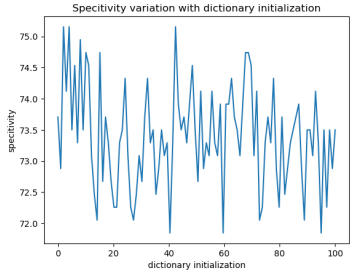

Figure 3: Variation of sensitivity, specificity and balanced accuracy with dictionary initialization for the ISIC dataset.

Through inspection of the figures it is clear that the dictionary initialization does introduce some significant variability in the final results. This variability is more apparent in the sensitivity with values in the range $[45.29, 53.85]$, while the specificity is in the range $[71.84, , 75.16]$, which translates into the balanced accuracy being in the range $[59.41, 63.57]$.

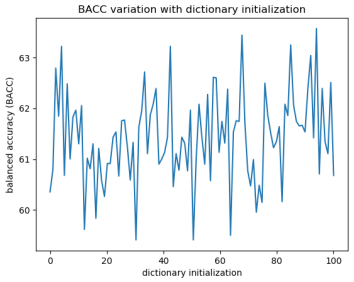

Figure 4: Variation of sensitivity, specificity and balanced accuracy with dictionary initialization for the ISIC dataset.

ISIC and EDRA

We have treated the ISIC and EDRA datasets as two separate datasets and reported results for all of the methods in both datasets. Here, and to ascertain the influence of the training data in the final result, the EDRA dataset is merged with the training and validation images of the ISIC dataset. This augmented set is then used to train the model described in section **??** and evaluated on the test set of the ISIC dataset. The results are presented in table 2.

Table 2: Proposed system performance on ISIC test set when trained with a mergure of the EDRA dataset and the ISIC training and validation sets.

Training Set	SE (%)	SP (%)	*BACC* (%)
ISIC	$56,41$	$71,43$	$63,50$
Augmented set	$58,12$	$71,22$	$64,67$

The addition of the EDRA images to the training set resulted in an improvement of $1,17\%$ with respect to the same system trained only with the ISIC training and validation sets. This improvement also extended to the ROC AUC of the system, achieving, where it was more pronounced, achieving $67,41$. Even though there was an improvement in the BACC, this value is still low, probably due to the EDRA images not being representative of the ISIC test set. This just goes to show the difficulty of the ISIC test set.

Conclusion

This work focused on the analysis of several methods and algorithms based on sparse representations.

An initial baseline system was proposed to tackle the problem of melanoma classification. This system made use of handcrafted features such as color and gradient histograms represented by sparse coding using over-complete dictionaries. The baseline system is topped by a late fusion of support vector machines that performs the final classification.

The notion of discriminative dictionaries is introduced, as well as a couple of methods that make use of them. Their integration is the baseline system is also discussed. The chapter ends by introducing hierarchical clustering applied to the dictionary atoms with the objective of cutting inter-class common atoms.

Finally the use of deep learning for the problem of melanoma detection is explored. Transfer learning is made using a convolutional neural network pre-trained on the ImageNet dataset, namely the VGG19. Features are extracted from this network and are applied in the baseline system as yet another source of information for melanoma classification.

Two datasets were used, the ISIC and EDRA datasets. Promising results were achieved for both datasets, with different systems. The system presented in section 4.3 with deep features achieved a BACC of $63,50\%$ on the ISIC dataset and the system presented in section 4.1 achieved a BACC of $76,83\%$ for the EDRA dataset.

Future work

The final system obtained, even though it achieved promising results, is quite simple compared to some state-of-the-art methods for image classification which mainly make use of convolutional neural networks.

It would be interesting to test this system on other public image datasets to see how it performs in an area other than skin cancer detection. The application of hierarchical clustering to any dictionary could also be further researched, since it quite increased the capability of the system in such a difficult dataset as is the ISIC dataset. Its inclusion in more modern deep learning systems for feature pruning could also be considered and studied.

With the increasing complexity of deep learning models, with neural networks that need to learn millions of parameters, making its use not only memory, but also time consuming, methods like sparse representations and clustering, which cuts a lot of redundancies and therefore reduces memory and boosts efficiency, seem to be a promising direction in the near future.

Acknowledgements

I would like to thank Prof. Jorge Marques and Dr. Ana Catarina Barata for supervising my thesis and for everything they taught me and helped me with during this process.

References

[1] American Cancer Institute. https://www.cancer.org/cancer/skin-cancer.html. [Online; accessed 01-April-2019].

[2] ImageNet website. http://www.image-net.org/. [Online; accessed 03-April-2019].

[3] ISIC 2017 challenge homepage. https://challenge.kitware.com/#challenge/n/ISIC_2017%3A_Skin_Lesion_Analysis_Towards_Melanoma_Detection. [Online; accessed 06-April-2019].

[4] Q. Abbas, M. E. Celebi, and I. Fondn. Computer-aided pattern classification system for dermoscopy images. *Skin research and technology : official journal of International Society for Bioengineering and the Skin (ISBS) [and] International Society for Digital Imaging of Skin (ISDIS) [and] International Society for Skin Imaging (ISSI)*, 18:278–89, 08 2011.

[5] G. Argenziano, G. Fabbrocini, P. Carli, V. De Giorgi, E. Sammarco, and M. Delfino. Epiluminescence Microscopy for the Diagnosis of Doubtful Melanocytic Skin Lesions: Comparison of the ABCD Rule of Dermatoscopy and a New 7-Point Checklist Based on Pattern Analysis. *Archives of Dermatology*, 134(12):1563–1570, 12 1998.

[6] C. Barata, M. E. Celebi, and J. S. Marques. Improving dermoscopy image classification using color constancy. *IEEE J. Biomedical and Health Informatics*, 19(3):1146–1152, 2015.

[7] C. Barata, M. Figueiredo, M. E. Celebi, and J. Marques. Local features applied to dermoscopy images: Bag-of-features versus sparse coding. pages 528–536, 05 2017.

[8] C. Barata, M. Ruela, M. Francisco, T. Mendonça, and J. S. Marques. Two systems for the detection of melanomas in dermoscopy images using texture and color features. *IEEE Systems Journal*, 8(3):965–979, 2014.

[9] M. E. Celebi, H. A. Kingravi, B. Uddin, H. Iyatomi, Y. A. Aslandogan, W. V. Stoecker, and R. H. Moss. A methodological approach to the classification of dermoscopy images. *Comp. Med. Imag. and Graph.*, 31(6):362–373, 2007.

[10] M. E. Celebi and A. Zornberg. Automated quantification of clinically significant colors in dermoscopy images and its application to skin lesion classification. *IEEE Systems Journal*, 8(3):980–984, 2014.

[11] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 01 1995.

[12] R. Erol, M. Bayraktar, S. Kockara, S. Kaya, and T. Halic. Texture based skin lesion abruptness quantification to detect malignancy. *BMC Bioinformatics*, 18(14):51–60, 2017.

[13] H. Finch. Comparison of distance measures in cluster analysis with dichotomous data. *Journal of Data Science*, 3:85–100, 01 2005.

[14] H. Iyatomi, H. Oka, M. E. Celebi, M. Hashimoto, M. Hagiwara, M. Tanaka, and K. Ogawa. An improved internet-based melanoma screening system with dermatologist-like tumor area extraction algorithm. *Comp. Med. Imag. and Graph.*, 32(7):566–579, 2008.

[15] H. Iyatomi, H. Oka, M. E. Celebi, M. Tanaka, and K. Ogawa. Parameterization of dermoscopic findings for the internet-based melanoma screening system. pages 189 – 193, 05 2007.

[16] S. Kong and D. Wang. A dictionary learning approach for classification: Separating the particularity and the commonality. In *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part I*, pages 186–199, 2012.

[17] K. Korotkov and R. Garcia. Computerized analysis of pigmented skin lesions: A review. 56, 10 2012.

[18] J. Mairal. SPAMS library. http://spams-devel.gforge.inria.fr/. [Online; accessed 05-March-2018].

[19] J. Mairal, F. R. Bach, and J. Ponce. Task-driven dictionary learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(4):791–804, 2012.

[20] K. Matsunaga, A. Hamada, A. Minagawa, and H. Koga. Image classification of melanoma, nevus and seborrheic keratosis by deep neural network ensemble. 03 2017.

[21] A. Menegola, M. Fornaciali, R. Pires, S. E. F. de Avila, and E. Valle. Towards automated melanoma screening: Exploring transfer learning schemes. *CoRR*, abs/1609.01228, 2016.

[22] K. Mllersen, M. Zortea, K. Hindberg, T. Schopf, S. Skrvseth, and F. Godtliebsen. *Improved Skin Lesion Diagnostics for General Practice by Computer-Aided Diagnostics*, pages 247–292. 09 2015.

[23] S. Pathan, K. G. Prabhu, and P. C. Siddalingaswamy. Techniques and algorithms for computer aided diagnosis of pigmented skin lesions - a review. *Biomed. Signal Proc. and Control*, 39:237–262, 2018.

[24] V. Pomponiu, H. Nejati, and N. Cheung. Deepmole: Deep neural networks for skin mole lesion classification. In *2016 IEEE International Conference on Image Processing, ICIP 2016, Phoenix, AZ, USA, September 25-28, 2016*, pages 2623–2627, 2016.

[25] M. Rastgoo, R. García, O. Morel, and F. Marzani. Automatic differentiation of melanoma from dysplastic nevi. *Comp. Med. Imag. and Graph.*, 43:44–52, 2015.

[26] M. Rastgoo, G. Lemaitre, O. Morel, J. Massich, R. García, F. Mériaudeau, F. Marzani, and D. Sidibé. Classification of melanoma lesions using sparse coded features and random forests. In *Medical Imaging 2016: Computer-Aided Diagnosis, San Diego, California, United States, 27 February - 3 March 2016*, page 97850C, 2016.

[27] S. Seidenari, G. Pellacani, and C. Grana. Colors in atypical nevi: A computer description reproducing clinical assessment. *Skin research and technology : official journal of International Society for Bioengineering and the Skin (ISBS) [and] International Society for Digital Imaging of Skin (ISDIS) [and] International Society for Skin Imaging (ISSI)*, 11:36–41, 03 2005.

[28] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[29] R. Stanley, W. Stoecker, and R. Moss. A relative color approach to color discrimination for malignant melanoma detection in dermoscopy images. *Skin research and technology : official journal of International Society for Bioengineering and the Skin (ISBS) [and] International Society for Digital Imaging of Skin (ISDIS) [and] International Society for Skin Imaging (ISSI)*, 13:62–72, 03 2007.

[30] A. Steiner, H. Pehamberger, and K. Wolff. In vivo epiluminescence microscopy of pigmented skin lesions. ii. diagnosis of small pigmented skin lesions and early detection of malignant melanoma. *Journal of the American Academy of Dermatology*, 17(4):584 – 591, 1987.

[31] W. Stoecker, W. Weiling Li, and R. Moss. Automatic detection of asymmetry in skin tumors. *Computerized medical imaging and graphics : the official journal of the Computerized Medical Imaging Society*, 16:191–7, 05 1992.

[32] W. Stolz, A. Riemann, A. B. Cognetta, L. Pillet, W. Abmayer, D. Holzel, P. Bilek, F. Nachbar, M. Landthaler, and O. Braun-Falco. Abcd rule of dermatoscopy: A new practical method for early recognition of malignant melanoma. *European Journal of Dermatology*, 4:521–527, 01 1994.

[33] M. Wang, W. Xu, and A. Tang. On the performance of sparse recovery via l_p-minimization ($0 <= p <= 1$). *IEEE Trans. Information Theory*, 57(11):7255–7278, 2011.

[34] O. Yim and K. Ack Baraly. Hierarchical cluster analysis: Comparison of three linkage measures and application to psychological data. *The Quantitative Methods for Psychology*, 11:8–21, 02 2015.

[35] Q. Zhang and B. Li. Discriminative K-SVD for dictionary learning in face recognition. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, pages 2691–2698, 2010.

[36] M. Zortea, T. R. Schopf, K. Thon, M. Geilhufe, K. Hindberg, H. M. Kirchesch, K. Møllersen, J. Schulz, S. O. Skrøvseth, and F. Godtliebsen. Performance of a dermoscopy-based computer vision system for the diagnosis of pigmented skin lesions compared with visual evaluation by experienced dermatologists. *Artificial Intelligence in Medicine*, 60(1):13–26, 2014.