Organ identification on shrimp histological images: A comparative study considering CNN and feature engineering

M. Mendieta[1]; F. Panchana[2]; B. Andrade[2]; B. Bayot[2]; C. Vaca[1]; B.X. Vintimilla[1]; D. G. Romero[1]
ESPOL Polytechnic University, Escuela Superior Politécnica del Litoral, ESPOL
Centro de Investigación desarrollo e Innovación de Sistemas Computacionales (CIDIS)[1]
Centro Nacional de Acuicultura e Investigaciones Marinas (CENAIM)[2]
Campus, Gustavo Galindo Km. 30.5 Vía Perimetral, P.O. Box 09-01-5863, Guayaquil, Ecuador.
{mvmendie;panchanf;beandrad;bbayot;cvaca;boris.vintimilla;dgromero}@espol.edu.ec

***Abstract*—The identification of shrimp organs in biology using histological images is a complex task. Shrimp histological images poses a big challenge due to their texture and similarity among classes. Image classification by using feature engineering and convolutional neural networks (CNN) are suitable methods to assist biologists when performing organ detection. This work evaluates the Bag-of-Visual-Words (BOVW) and Pyramid-Bag-of-Words (PBOW) models for image classification leveraging big data techniques; and transfer learning for the same classification task by using a pre-trained CNN. A comparative analysis of these two different techniques is performed, highlighting the characteristics of both approaches on the shrimp organs identification problem.**

***Index Terms*—Aquaculture, histology, organ identification, CNN, transfer learning, feature extraction, fine-tuning, big data.**

I. INTRODUCTION

Nowadays, different tools have been developed to categorize a variety of applications in biology such as: registration of known animal species [1]; categorization of genus [2]; cell segmentation and cell counting from histological images [3], amongst others [4] [5] [6]. Complementarily, aquaculture plays a very important role in the world food security and global economic development [7]. It is one of the fastest growing food production sectors in the world due to the increased demand for fish and crustacean [8] [9]. Species cultivated in aquaculture include crustaceans, mollusks and fishes [10]. However, the major constraints to the growth of the aquaculture sector are infectious diseases, which cause great economic losses [11]. Due to diseases, aquaculture production loses its economic value due to increased mortality, growth problems and even a bad taste and changes in appearance [12]. On the other hand, the variability of some water quality parameters beyond acceptable limits can also cause slow growth and stress that contributes to the emergence of diseases [13].

Shrimp farming is one of the main aquaculture activities world wide and in constant growth [14]. However, viral and bacterial diseases are major constraints on their production, becoming a threat to their sustainability. Histology is one of the main tools for organ detection and disease diagnosis, and it is especially important when molecular tools are not available for the specific detection of a pathogen.

The design of a suitable method for automatic recognition of organs demands a multi-disciplinary effort from different areas of research such as life sciences, computer science and engineering. An histology-based automatic identification approach in aquaculture requires, at the beginning, to identify the shrimp organs before attempting to discover the presence of a particular disease. Organ identification is not a trivial task, even for humans, especially by using histological images obtained with a 40x-zoom microscope, which are the kind of images used in this work [15].

The present work uses a dataset created at ESPOL University. It consist of 2.535 histological images of Penaeus (Litopenaeus) vannamei shrimp organs, categorized by disease (white spot disease - WSD and vibriosis, caused by white spot syndrome virus - WSSV and pathogenic bacteria species of the family Vibrionaceae) and taken at 40X microscope zoom. The dataset distributed in 11 shrimp organs (classified as healthy, WSD and vibriosis) is detailed in Table I.

Table I: Dataset used in the paper: 11 organs classified as healthy, WSD and Vibriosis

Ord	Organ	Healthy	WSD	Vibriosis	Total
1	Gills	146	71	70	287
2	Connective stomach	151	70		221
3	Body epithelium	144	70	80	294
4	Stomach epithelium	141	70		211
5	Intestine epithelium	147	70		217
6	Nervous Cord	146	70		216
7	Hematopoietic tissue	170	70		240
8	Hepatopancreas	147		70	217
9	Muscle	164		62	226
10	Linfoid organ	126	70		196
11	Pleopods	140	70		210
	Total	**1622**	**631**	**282**	**2535**

A histological image has a particular structure composed with few colors and different shapes of borders and textures. Furthermore, when these images are zoomed-in with the

microscope, different organs could merge together in the same image, which increases the complexity of the problem at hand.

Over the past few years, deep CNNs have revolutionized large-scale image recognition and classification. Virtually all of today's high achieving algorithms and architectures for image classification and recognition make use of deep CNN architectures in some way [16] [17] [18]. These advances have been made possible by large public image repositories and the use of high performance GPUs. CNNs are a set of layers in which each layer performs a non-linear transformation function learned from a labeled set of data. However, the deep learning approach requires datasets with large sizes, which increases at the same time costs and efforts that are hard to sustain indefinitely when working with histological images.

The purpose of this paper is to compare the use of feature engineering, with big data techniques, and deep learning in aquaculture, to identify shrimp organs by using histological images. The paper is structured as follows: In part II the methodology used during the experiments is presented. Part III describes the implementation of all the experiments conducted during the research. Part IV shows the results. Part V includes discussion and future works and finally, in Part VI the conclusions are presented.

II. Methodology

The image classification pipeline for deep learning and feature engineering is comprised of 5 steps as depicted in Fig. 1 (step 3 does not apply to deep learning). Each step is defined as follows:

- **Step 1: Structuring the initial dataset.** This step involves the images themselves as well as the labels associated with each image. It is important that the number of images for each category is fairly uniform or balanced, in order to avoid any bias during the training step. For datasets too big to fit in memory, or for more efficient methods to train deep networks, it is more convenient to store the dataset in an HDF5 file, which is a binary format that can be accessed in slices or batches in a NumPy-like fashion.
- **Step 2: Splitting the dataset** into two (optionally three) parts.
- **Step 3: Extracting features** to abstractly quantify and represent each image. Common choices of features include: LBP, HOG, SIFT, amongst others [19]. This step does not apply for a deep learning approach.
- **Step 4: Training the classification model**. For feature engineering, this step depends on each individual algorithm such as: Support Vector Machine (SVM), Random Forest, Decision Trees, etc. For deep learning, we apply a form of gradient descent, or other advanced optimization techniques: Adagrad [20], Adadelta [21], Adam [22], and Nadam [23].
- **Step 5: Evaluating** the classifier or the network using metrics such as: Precision, recall, F1 score, confusion matrix, among others.

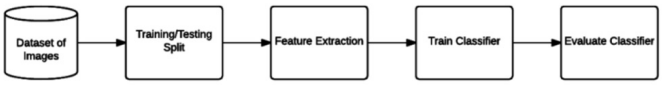

Figure 1: Image Classification Pipeline

III. Implementation

All the experiments were carried out on a workstation computer with GPU capability (Titan X, 12 GB RAM). The spark configuration used in this work includes python programming interface or pyspark, standalone mode on a single machine, and interactive mode using jupyter notebook.

A. Feature Engineering Experiments

A total of 18 experiments were carried out in order to classify shrimp organs, using different configurations of SURF keypoint detector and RootSIFT feature extractor, and Local Binary Patterns. From all the experiments, 16 of them were implemented using the BOVW model y the remaining two with the PBOW model. As a matter of simplicity, the results of the eight experiments achieving over 80% accuracy are depicted in Table II and Table III. The entire dataset was splitted in 80% for training and 20% for testing. The classifier used in all the experiments, as well as the clustering algorithm used to construct the vocabulary of words, was SVM and K-means respectively.

The feature extraction process is inherently a task that can be made parallel, it was performed leveraging Hadoop Streaming using Python 3.5, as well as using well-know libraries for computer vision and machine learning such as: scikit-image, scikit-learn, sklearn, and Opencv 3.3. After running the Hadoop Streaming job for the SURF keypoint detector and RootSIFT descriptor, we ended up with the total feature vectors with 128-dim, distributed in over 150 part files totalling sizes over 15 GB that correspond to the entire features extracted from the original dataset. The amount of feature vectors depends on the Hessian Threshold used with the SURF keypoint detector, ranging from 250.000 to approximately 5.500.000 for all the experiments.

Clustering the features to form a visual vocabulary was performed by leveraging *pyspark* and *MLlib*, whose input files were the features extracted in the preceding step, and the resulting output were the visual words required to construct the bag-of-words in the next step. The K-Means model implemented in *MLlib* requires only a single feature column with float values, this is accomplished with the *VectorAssembler* function. Before implementing the K-means algorithm, it is required to find the best value of K that optimizes the model. This task was accomplished by using the *KMeans()* object, and the methods *fit* and *computeCost* for 13 different values of K (500, 1000, 1300, 1500, 1700, 2000, 2500, 3000, 3500, 4000, 5000, 7500, 10000). The best value of K falls between 3000 to 4000, as seen in Figure 2. This was the longest task performed during step 2 of the bovw model.

Table II: Precision, Recall and F1-score obtained in the experiments for each of the 11 organs using Feature Engineering

Ord	Organ	SURF_RootSIFT_k500			SURF_RS_HT400_k1500			SURF_RS_HT400_k2500			SURF_RootSIFT_k2000		
		Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
1	Gills	0.81	0.96	0.88	0.83	0.91	0.87	0.88	0.90	0.89	0.83	0.96	0.89
2	Connective stomach	0.89	0.98	0.93	0.96	0.95	0.96	0.94	0.98	0.96	1.00	0.98	0.99
3	Body epithelium	0.71	0.70	0.70	0.92	0.62	0.74	0.80	0.77	0.78	0.68	0.80	0.74
4	Stomach epithelium	0.67	0.63	0.65	0.80	0.74	0.76	0.78	0.74	0.76	0.86	0.59	0.70
5	Intestine epithelium	0.78	0.86	0.82	0.88	0.87	0.87	0.85	0.94	0.89	0.88	0.86	0.87
6	Nervous Cord	0.80	0.89	0.84	0.69	1.00	0.82	0.84	0.98	0.91	0.85	0.89	0.87
7	Hematopoietic tissue	0.79	0.80	0.80	0.79	0.77	0.78	0.87	0.80	0.83	0.69	0.92	0.79
8	Hepatopancreas	0.88	0.90	0.89	0.81	0.88	0.85	0.91	0.95	0.93	0.79	0.90	0.84
9	Muscle	0.95	0.77	0.85	0.94	0.81	0.87	0.94	0.73	0.83	0.97	0.71	0.82
10	Linfoid organ	0.72	0.82	0.77	0.75	0.77	0.76	0.70	0.86	0.77	0.85	0.76	0.81
11	Pleopods	0.73	0.51	0.60	0.67	0.72	0.70	0.72	0.62	0.67	0.68	0.53	0.60
	Total	**0.80**	**0.80**	**0.79**	**0.83**	**0.82**	**0.81**	**0.84**	**0.84**	**0.84**	**0.82**	**0.81**	**0.81**

Table III: Precision, Recall and F1-score obtained in the experiments for each of the 11 organs using Feature Engineering

Ord	Organ	SURF_RootSIFT_k2500			SURF_RootSIFT_k3000			SURF_RootSIFT_k5000			SURF_RootSIFT_k10000		
		Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
1	Gills	0.82	0.91	0.87	0.88	0.93	0.91	0.85	0.96	0.90	0.88	0.93	0.91
2	Connective stomach	0.94	0.96	0.95	0.89	0.98	0.93	0.89	0.98	0.93	0.96	0.96	0.96
3	Body epithelium	0.79	0.73	0.76	0.63	0.89	0.74	0.73	0.73	0.73	0.72	0.73	0.73
4	Stomach epithelium	0.74	0.56	0.64	0.86	0.61	0.71	0.70	0.63	0.67	0.71	0.66	0.68
5	Intestine epithelium	0.88	0.86	0.87	0.83	0.93	0.88	0.91	0.74	0.82	0.86	0.88	0.87
6	Nervous Cord	0.88	0.80	0.83	0.82	0.91	0.86	0.89	0.73	0.80	0.61	0.98	0.75
7	Hematopoietic tissue	0.84	0.79	0.81	0.74	0.92	0.82	0.81	0.77	0.79	0.95	0.57	0.71
8	Hepatopancreas	0.65	0.98	0.78	0.90	0.86	0.88	0.93	0.90	0.92	0.87	0.79	0.82
9	Muscle	0.95	0.77	0.85	0.98	0.79	0.87	0.89	0.81	0.85	0.93	0.79	0.85
10	Linfoid organ	0.66	0.97	0.79	0.87	0.71	0.78	0.58	0.95	0.72	0.53	0.97	0.69
11	Pleopods	0.80	0.62	0.70	0.86	0.45	0.59	0.72	0.64	0.68	0.82	0.43	0.57
	Total	**0.82**	**0.81**	**0.81**	**0.83**	**0.82**	**0.81**	**0.81**	**0.80**	**0.80**	**0.81**	**0.78**	**0.77**

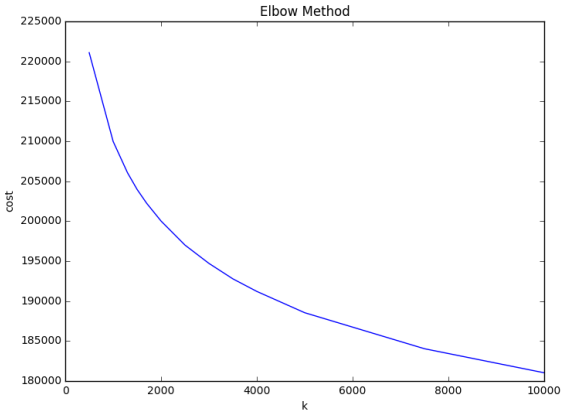

Figure 2: Elbow method for different values of K

B. Deep Learning Experiments

We implemented 8 experiments using three deep learning techniques: 1) training *MiniVGGNet*, an *end-to-end* shallow network, 2) treating networks as feature extractors using a *pre-trained* VGG16 on Imagenet, and 3) *Fine-tuning* VGG16 with our own dataset. The exact configuration of each experiment is detailed below, and four most accurate results are shown in Table IV:

- **Experiment #1 end-to-end networks:** Training *MiniVGGNet* with Batch Normalization, No data augmentation, and resizing the images to 64 x 64.
- **Experiment #2 end-to-end networks:** Similar to experiment #1, but resizing the images to 96 x 96.
- **Experiment #3 networks as feature extractors:** The network used is VGG16 from Keras trained on Imagenet. We use the features from the last layer previous to the fully connected, and fed this feature vector to a Logistic Regression classifier. Images were resized to 224 x 224, data augmentation techniques were used as well.
- **Experiment #4 *Fine-tuning:*** Using the same network as in the previous experiment, but placing a new layer after layer 18, with the following architecture: INPUT – FC – RELU – DO – FC – OUTPUT. The training was performed using data augmentation, RMSprop optimization, and the images were resized to 224 x 224. After the *warm-up* training phase, we trained the whole network from layer 15.
- **Experiment #5 *Fine-tuning:*** Similar to experiment #4 but this time the *fine-tuning* begins at layer 10.
- **Experiment #6 *Fine-tuning:*** Similar to Experiment #4 but this time using a *learning rate scheduler* decreasing the learning rate by 0.25 every 10 epochs. The training was performed with SGD with momentum and nesterov acceleration from layer 15.
- **Experiment #7 *Fine-tuning:*** Similar to Experiment #6 but the final training phase was performed from layer 10.
- **Experiment #8 *Fine-tuning:*** Similar to Experiment #6 with a learning rate decay of 0.5 every 10 epochs.

Table IV: Precision, Recall and F1-score obtained in the experiments for each of the 11 organs using Deep Learning techniques

Ord	Organ	Experiment #4			Experiment #5			Experiment #6			Experiment #8		
		Prec.	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**	**Prec.**	**Rec.**	**F1**
1	Gills	0.95	1.00	0.97	0.99	0.96	0.97	0.93	1.00	0.96	0.93	1.00	0.96
2	Connective stomach	0.98	0.95	0.96	0.98	0.93	0.95	1.00	0.91	0.95	1.00	0.91	0.95
3	Body epithelium	0.92	0.92	0.92	0.91	0.94	0.92	0.89	0.87	0.88	0.92	0.90	0.91
4	Stomach epithelium	0.89	0.82	0.85	0.95	0.87	0.90	0.86	0.85	0.86	0.88	0.85	0.86
5	Intestine epithelium	0.92	0.89	0.90	0.98	0.94	0.96	0.92	0.89	0.90	0.96	0.92	0.94
6	Nervous Cord	1.00	0.97	0.98	0.97	1.00	0.98	1.00	0.95	0.97	0.98	0.97	0.97
7	Hematopoietic tissue	0.82	0.98	0.89	0.88	0.98	0.93	0.81	0.96	0.88	0.83	0.98	0.90
8	Hepatopancreas	0.98	0.98	0.98	0.96	1.00	0.98	0.96	0.96	0.96	0.98	0.98	0.98
9	Muscle	0.94	0.94	0.94	0.96	0.98	0.97	0.94	0.96	0.95	0.98	0.96	0.97
10	Linfoid organ	0.93	0.80	0.86	0.96	0.92	0.94	0.93	0.82	0.87	0.91	0.84	0.88
11	Pleopods	0.84	0.90	0.87	0.90	0.92	0.91	0.82	0.85	0.84	0.88	0.90	0.89
	Total	**0.95**	**0.95**	**0.93**	**0.92**	**0.92**	**0.95**	**0.92**	**0.91**	**0.91**	**0.93**	**0.93**	**0.93**

IV. Results

The evidence shown in Tables II and III demonstrates that for histological images the BOVW model is more efficient than the PBOW. Moreover, SURF and RootSIFT are more robust than LBP's for this type of applications. The model with the highest accuracy, 84 %, was SURF _RootSIFT _HT400 _k2500, which is a combination of the SURF keypoint detector, RootSIFT feature extractor, a Hessian Threshold of 400 for the SURF algorithm, and a cluster of words equals to 2500.

Previously in this paper it was mentioned that one of the main drawbacks when working with histological images is the interclass similarity among some classes. One way to evidence this similarity is by graphically showing the 16 most similar image patches relevant to each codeword in the vocabulary (i.e k=1500, k3000, etc), together with their corresponding class label. Feature vectors with a smaller distance to a given visual word are considered to be more relevant. For instance, for the experiment SURF _RootSIFT _k1500, the visualization of the word #259 in Fig. 3 is relevant enough to correctly discriminate the class muscle. On the other hand, the visualization of the word #355 in Fig. 4 is not discriminative enough, so the patches correspond to many different labels.

This situation arises because the BOVW model implemented in this paper uses the concept of "Term frequency", or simply the number of times a visual word appears in an image. Visual words that are more common across all images in a dataset (i.e., less informative) will have more entries in the histogram, whereas visual words that are rare (i.e., more informative) will have less entries in the bovw histogram, which is not the desired effect.

The results shown in IV demonstrates that for histological images, shallow networks such as *MiniVGGNet* and transfer learning in the form of feature extractors do not perform well. On the other hand, *fine-tuning* obtained accuracies over 90%, being the most accurate model the one implemented in experiment #4 with 95% accuracy, an improvement of 11% in comparison to the best model in the feature engineering approach. Moreover, the patterns learned in the low-level layers of *pre-trained* deep network architectures on the Imagenet dataset, such as VGG, are discriminative enough to classify histological images in the higher-level layers.

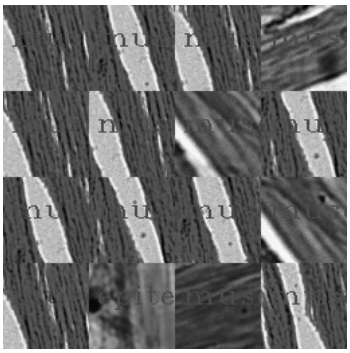

Figure 3: Discriminative codeword (259)

Figure 4: Non-discriminative codeword (355)

Table V shows the mean and standard deviation for the three classification metrics (precision, recall, and F1-score) computed in the experiments for each organ using feature engineering and deep learning. It is clear that in the case of feature engineering, the four organs with precision scores below 80% are: *body epithelium*, *stomach epithelium*, *linfoid organ*, and *pleopods*; whereas for deep learning, there are only two organs achieving less than 90% precision: *hematopoietic tissue*, and *pleopods*. Note that *pleopods* obtain the least precision in both approaches, so it might be neccessary to perform extra image pre-processing of this particular dataset before passing them through the classifier.

In order to spot overfitting during the training phase, the respective learning curves for the best models in

Table V: Mean and standard deviation for each organ with the results obtained from the 18 experiments

Ord	Organ	Mean Prec.	Mean Rec.	Mean F1	Stdev Prec.	Stdev Rec.	Stdev F1	Mean Prec.	Mean Rec.	Mean F1	Stdev Prec.	Stdev Rec.	Stdev F1
1	Gills	0.85	0.93	0.89	0.03	0.02	0.02	0.95	0.99	0.97	0.03	0.02	0.01
2	Connective stomach	0.93	0.97	0.95	0.04	0.01	0.02	0.99	0.93	0.95	0.01	0.02	0.00
3	Body epithelium	0.75	0.75	0.74	0.09	0.08	0.02	0.91	0.91	0.91	0.01	0.03	0.02
4	Stomach epithelium	0.77	0.65	0.70	0.07	0.07	0.05	0.90	0.85	0.87	0.04	0.02	0.02
5	Intestine epithelium	0.86	0.87	0.86	0.04	0.06	0.03	0.95	0.91	0.93	0.03	0.02	0.03
6	Nervous Cord	0.80	0.90	0.84	0.10	0.09	0.05	0.99	0.97	0.98	0.02	0.02	0.01
7	Hematopoietic tissue	0.81	0.79	0.79	0.08	0.11	0.04	0.84	0.98	0.90	0.03	0.01	0.02
8	Hepatopancreas	0.84	0.90	0.86	0.09	0.06	0.05	0.97	0.98	0.98	0.01	0.02	0.01
9	Muscle	0.94	0.77	0.85	0.03	0.04	0.02	0.96	0.96	0.96	0.02	0.02	0.01
10	Linfoid organ	0.71	0.85	0.76	0.12	0.10	0.04	0.93	0.86	0.89	0.92	0.08	0.04
11	Pleopods	0.75	0.57	0.64	0.07	0.10	0.05	0.88	0.89	0.88	0.06	0.03	0.03

both approaches were plotted, this is SURF _RootSIFT _k2500, whose BOVW sizes were (2524, 2500), for feature engineering, and Experiment #4 for deep learning.In the first case, we note that the amount of training samples in the dataset were sufficient enough for the model to learn without any bias or variance. In the latter, we notice some signs of overfitting past epoch 45; however, by using Keras *callback* functionality, we checkpointed the best model and serialized it to disk for further evaluation.

Table VI shows the the duration of each task required to perform step 2 of the bovw model using pyspark, which accounts for more than 2 1/2 days. Optimizing the best value of K is the longest task followed by extracting the cluster centers, both tasks are candidates to be performed in the cloud to speed up the process.

Table VI: Duration of Step 2 of the BOVW model using PySpark

Ord	Task	Days	Hours	Minutes	Seconds
1	Load files			6	43
2	Print shape			4	11
3	Optimize K	1	23	6	52
4	Cluster centers		17	27	6
	Total	**2**	**16**	**44**	**52**

The performance obtained by the best models in both approaches was compared with the results obtained in a similar experiment presented in [24], where the authors used feature extraction techniques with SIFT as both keypoint detector and feature extractor, the BOVW model for image classification, and a SVM classifier; and for deep learning, they used fine-tuning with a *pre-trained* AlexNet architecture. The comparison is shown in Table VII using the F1-score metric.

For the feature engineering approach, it is clearly observed that the combination of SURF - RootSIFT drastically improves the performance over SIFT-SIFT in the following organs: *connective stomach*, *nervous cord*, *hepatopancreas*, and *muscle*. In the case of *Gills* the performance barely improves, and for *body epithelium* there is a decrease in performance. In the case of deep learning, our proposed approach of *fine-tuning* a *pre-trained* VGG16 network architecture outperforms the *AlexNet* architecture in 2% overall.

Table VII: F1 score comparison with results obtained in a similar work using (SIFT-SIFT) and DNN Alexnet with our proposed approach (SURF-RootSIFT) and DNN VGG16

ORGAN	SIFT-SIFT	SURF-RootSIFT	DNN Alexnet	DNN VGG
Connective stomach	0.693	0.96	0.934	0.96
Nervous cord	0.478	0.91	0.995	0.98
Hepatopancreas	0.706	0.93	0.934	0.98
Body epithelium	0.885	0.78	0.997	0.92
Muscle	0.380	0.83	0.987	0.94
Gills	0.864	0.89	0.993	0.97
Total		**0.84**	**0.93**	**0.95**

V. Future Work

This work considered the use of 3 algorithms for keypoint detection and feature extraction (SURF, RootSIFT, and LBP). Other algorithms like Histogram of Oriented Gradients (HOG), color histograms, Good Features to Track (GFFT), STAR, or Harris could be implemented in the BOVW model, even an ensemble of feature vectors,in conjunction with a technique of feature selection could improve the model performance.

A more robust method to determine the similarity between classes after obtaining the BOVW representation (histogram) is the chi-square distance, a very popular metric used to compare histogram distributions. This metric will allow us to determine which classes are more similar to one another, in order to implement extra image pre-processing techniques to ensure that these classes are different.

The use of Term frequency-inverse document frequency, or simply *tf-idf* for short, as the input to the BOVW model for image classification, could improve the results due to the presence of words in the vocabulary that are not discriminative enough, and therefore do not contribute with relevant information to the classifier.

An increase in performance with Deep learning could be obtained by *fine-tuning* other state of the art *pre-trained* network architectures, such as: ResNet50, VGG19, Inception V3, Xception and so forth; training a new model from scratch or performing an ensemble of networks. The last two approaches would require more computing power and a bigger dataset. On the other hand, including infrared spectrum as additional information during image capturing process is being considered [16].

VI. Conclusions

A comparative study was conducted in order to evaluate two approaches of featuring engineering using the BOVW model for the automatic detection of 11 shrimp organs from histological images. A combination of SURF keypoint detector and RootSIFT keypoint extractor provided better classification performance than SIFT - SIFT, and even LBP, both using the BOVW model. Moreover, the BOVW model overcomes the PBOW model for the classification of organs from histological images. There are 3 classes of organs (linfoid organ, pleopods, and epithelium stomach), that regardless the method used, always yield low classification results, which led to apply a different technique such as TF-IDF. The size of the dataset is big enough to implement feature engineering techniques without overfitting, so the technique is suitable to be considered on a limited input data. However, the results obtained by using CNN were better on relatively small CNN models (7 layers on miniVGGNet), this is also favorable in terms of portability and complexity.

ACKNOWLEDGMENT

This work has been supported by Escuela Superior Politécnica del Litoral (ESPOL), within the research project "Pattern recognition: Study cases in agriculture and aquaculture" M1-DI-2015 and by Secretaria de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT), in the framework of the PIC-14-CENAIM-003 project: "Desarrollo e implementación de métodos de control y prevención de enfermedades en especies acuáticas de uso comercial y uso potencial en maricultura y repoblación".

References

[1] E. Nava, E. I. Villar, M. C. Clemente, J. Rey, A. García, L. Fernández-Peralta, C. G. Piñeiro, and P. Otero, "Digital imaging tool to enhance otolith microstructure for estimating age in days in juvenile and adult fish," *IEEE Journal of Oceanic Engineering*, vol. 43, no. 1, pp. 48–55, Jan 2018.

[2] J. A. Carvajal, D. G. Romero, and A. D. Sappa, "Fine-tuning based deep convolutional networks for lepidopterous genus recognition," in *Iberoamerican Congress on Pattern Recognition*. Springer, 2016, pp. 467–475.

[3] Z. Wang, "A new approach for segmentation and quantification of cells or nanoparticles," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 962–971, June 2016.

[4] K. Harris, T. L. Parsons, U. Z. Ijaz, L. Lahti, I. Holmes, and C. Quince, "Linking statistical and ecological theory: Hubbell unified neutral theory of biodiversity as a hierarchical dirichlet process," *Proceedings of the IEEE*, vol. 105, no. 3, pp. 516–529, March 2017.

[5] Z. Z. Chen, Q. Feng, C. Shen, J. Wang, and L. Wang, "Algorithms for pedigree comparison," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 2, pp. 422–431, March 2018.

[6] R. Singh, R. Beasley, T. Long, and C. R. Caffrey, "Algorithmic mapping and characterization of the drug-induced phenotypic-response space of parasites causing schistosomiasis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 15, no. 2, pp. 469–481, March 2018.

[7] FAO, "State of the world fisheries and aquaculture: Opportunities and challenges," 2014. [Online]. Available: https://books.google.com.ec/books?id=-27boAEACAAJ

[8] R. P. Subasinghe, "Epidemiological approach to aquatic animal health management: opportunities and challenges for developing countries to increase aquatic production through aquaculture," *Preventive Veterinary Medicine*, vol. 67, no. 2, pp. 117–124, 2005.

[9] S. B. Chandanapalli, S. Reddy *et al.*, "Design and deployment of aqua monitoring system using wireless sensor networks and iar-kick," *Journal of Aquaculture Research & Development*, vol. 2014, 2014.

[10] A. G. Murray and E. J. Peeler, "A framework for understanding the potential for emerging diseases in aquaculture," *Preventive veterinary medicine*, vol. 67, no. 2, pp. 223–235, 2005.

[11] S. S. De Silva, "A global perspective of aquaculture in the new millennium," in *Technical Proceedings of the Conference on Aquaculture in the Third Millennium*. Network of Aquaculture Centres in Asia-Pacific, 2000, pp. 431–459.

[12] K. D. Lafferty, C. D. Harvell, J. M. Conrad, C. S. Friedman, M. L. Kent, A. M. Kuris, E. N. Powell, D. Rondeau, and S. M. Saksida, "Infectious diseases affect marine fisheries and aquaculture economics," *Annual review of marine science*, vol. 7, pp. 471–496, 2015.

[13] C. Anyadike, C. Mbajiorgu, and G. Ajah, "Aquacultural system management tool (aquasmat) i: Model development," *Aquacultural Engineering*, vol. 69, pp. 60–77, 2015.

[14] Fisheries and A. Department, "Shrimp, increased production of farmed shrimp leads to improved international trade," *GLOBEFISH Highlights*, pp. 13–15, 2017.

[15] M. Mendieta and D. Romero, "A cross-modal transfer approach for histological images: A case study in aquaculture for disease identification using zero-shot learning," in *Ecuador Technical Chapters Meeting (ETCM), 2017 IEEE*. IEEE, 2017, pp. 1–6.

[16] P. L. Suárez, A. D. Sappa, and B. X. Vintimilla, "Cross-spectral image patch similarity using convolutional neural network," in *2017 IEEE International Workshop of Electronics, Control, Measurement, Signals and their Application to Mechatronics (ECMSM)*, May 2017, pp. 1–5.

[17] J. Hu, C. Xiao, L. Shen, Q. Xie, X. Chen, and H. Han, "Automatical detecting and connecting the mitochondria from the serial em images," in *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, Aug 2017, pp. 1632–1637.

[18] H. Garud, S. P. K. Karri, D. Sheet, J. Chatterjee, M. Mahadevappa, A. K. Ray, A. Ghosh, and A. K. Maity, "High-magnification multi-views based classification of breast fine needle aspiration cytology cell samples using fusion of decisions from deep convolutional networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017, pp. 828–833.

[19] P. Ricaurte, C. Chilán, C. A. Aguilera-Carrasco, B. X. Vintimilla, and A. D. Sappa, "Feature point descriptors: Infrared and visible spectra," *Sensors*, vol. 14, no. 2, pp. 3690–3701, 2014.

[20] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[21] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] T. Dozat, "Incorporating nesterov momentum into adam," 2016.

[24] Y. B. P. Romero, Guale, "Evaluación de técnicas de clasificación orientadas a la identificación automática de órganos del camarón a partir de imágenes histológicas," *15th LACCEI International Multi-Conference for Engineering, Education, and Technology*, 2017.