

Inside-out Vision for Treatment Identification in Dental Setup using Machine Learning

Shaheena Noor (✉ ms_snoor@yahoo.com)

Sir Syed University of Engineering & Technology

Humera Noor Minhas

4pilots

Muhammad Imran Saleem

Sir Syed University of Engineering & Technology

Vali Uddin

Sir Syed University of Engineering & Technology

Najma Ismat

Sir Syed University of Engineering & Technology

Research Article

Keywords: Inside-out Vision , Machine Learning , Multinomial Logistic Regression , Lazy Instance-based Learning (IBk) , Sequential Minimal Optimization (SMO) , Hoeffding Tree , Random Tree

Posted Date: June 8th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-539758/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Inside-out Vision for Treatment Identification in Dental Setup using Machine Learning

Dr. Shaheena Noor · Dr. Humera Noor
Minhas · Engr. Muhammad Imran
Saleem · Dr. Vali Uddin · Dr. Najma
Ismat

Received: date / Accepted: date

Abstract Smart clinics have gained much popularity due to the technological advancements in areas like computer vision. The recognition of objects and activities and overall perceiving the environment lies at the core of such systems. This is essential not just for the eco-independent systems, but also for Human-Machine-Interaction - specially in scenarios with small work-areas like dental treatment. In this paper, we compare a number of machine learning models (including Multinomial Logistic Regression, Lazy Instance-based Learning (IBk), Sequential Minimal Optimization (SMO), Hoeffding Tree and Random Tree) for robustly identifying dental treatments. We take the objects-focussed as input, which covers parameters like material, symptoms of the patient teeth and tools used by the dentist. We take advantage of the fact that the issue of identifying a particular treatment can be solved by recognizing the objects seen during an activity. We collected a dental dataset in-the-wild and ran our tests to find that integrating different parameters improves accuracy relative to using each one separately. However, we also noted that in certain cases using the symptoms stand-alone gave better results. Also, with respect to RMS

Dr. Shaheena Noor
Computer Engineering Department, Sir Syed University of Engineering & Technology
Tel.: +92-300-278066
E-mail: shanoor@ssuet.edu.pk

Dr. Humera Noor Minhas
CTO and Co-Founder, 4pilots, Jena - Germany humera.noor@gmail.com

Engr. Muhammad Imran Saleem
Computer Engineering Department, Sir Syed University of Engineering & Technology
isaleem@ssuet.edu.pk

Dr. Vali Uddin
Vice Chancellor, Sir Syed University of Engineering & Technology
vc@ssuet.edu.pk

Dr. Najma Ismat
Computer Engineering Department, Sir Syed University of Engineering & Technology
nismat@ssuet.edu.pk

error convergence, symptoms showed to have lower error compared to combined. Finally, we noticed that the combined approach led to longer build and test times for the machine learning models. This shows that in machine learning applications in general and in medical/dental applications in particular, adding more parameters does not always lead to improved results. Rather it depends on the ML tool used, the parameters considered and the data given as input.

Keywords Inside-out Vision · Machine Learning · Multinomial Logistic Regression · Lazy Instance-based Learning (IBk) · Sequential Minimal Optimization (SMO) · Hoeffding Tree · Random Tree

1 Introduction

Healthcare heavily relies on images processing for not only medical diagnosis, but also for Human Computer Interaction systems and smart robots. This requires, on the one hand, sophisticated image capturing setups and efficient machine learning models on the other. As the wearable technology becomes more pervasive and advanced, it has made it possible to perceive the environment from the point of view of the actor. This has changed the working paradigm a lot, because now the scene can be perceived from the actor's point of view thus giving hints into his cognitive progress, and also save us from installing fixed camera setups around the actor. This mode of data perception, also called as First Person Vision, allows to interpret complicated scenarios by breaking them into their constituent objects seen and handled by the actor.

The paper is organized as follows: In Section 2, we provide an overview to the literature review on inside-out vision and machine learning for computer vision applications. In Section 3, we elaborate our system based on treatment identification using inside-out vision and introduce the ML models used in our work. In Section 4, we present our dataset, provide details of the experiments and a discussion of the results. Finally, in Section 5, we conclude our findings.

2 Literature Review

2.1 Inside-out vision

Inside-out vision associates human eye movements with human mind. There is a lot of research on human working behavior and its connection to the eye movement which in fact follow our mind and thoughts [1],[2]. The concept of eye movement and trajectories was proposed by Yarbus [3], that became the basis for future research [4] [5],[6]. Even for the same setup, if the task is given with different goals, the eye movement trajectories and patterns will be different. Inside-out vision is different from the traditional wall mounted cameras. Over here, the actor itself wears the camera on shoulder, forehead or eyes etc. and observe the scenario from his point of view. On the other hand

traditional wall mounted cameras are used to observe the overall situation of the environment - hence referred to as outside-in vision. They observe the overall activity especially focusing on human activity and behavior.

Outside-in view is well-suited for identifying person recognition, current activities, predict future activities, observing human behavior and many more. But there are also issues related to it if we want to capture items that are far away from the camera or they are small in size. Therefore, missing these items from the environment reduce the accuracy in the overall performance. For such scenarios, inside-out vision cameras are much more reliable as they are easy to install and can provide minor details of the scene. In short, we can say that these inside-out vision cameras is easily worn by the person and it exactly focuses where the person is looking. So relevant information will be captured and other additional information will be discarded.

Research has been done on scenario detection using inside-out vision cameras to note objects and behaviors activities. In [7], the importance of eye movement when perceiving a scene was investigated. Similarly, in [8] gaze patterns are being observed during a scenario execution and their linkage with the task being performed. Object and activity recognition [9] using deep features are very much used when we talk about computer vision and machine learning. It has been checked that the eye movements are closely linked to the current environment, job or operation and this is why the objects seen aid in situations recognition. It is particularly helpful when identification needs to be achieved without large movements and when the interacting objects are in close proximity to each other. Inside-out vision is widely being used in smart home, identifying objects [10], locations, observing human behavior [11] researching on neuro system [4], predicting mental disorder and illness. According to Furnari et al. in [12] inside-out vision views are used to estimate the future actions based on the interactions with the objects.

2.2 Machine Learning for Understanding and Prediction

As the image capturing devices are becoming more pervasive and sophisticated, a lot of input that we get from the environment is visual - in the form of images and videos. Understanding these images to perceive the environment is, therefore, an interesting and challenging task. Many efforts from machine learning researchers are being put in computer vision and image understanding applications. This includes not only identifying the low-level features apparent from the visual data, e.g. recognizing objects and activities, but also to analyse them for scene understanding and behavior analysis. This automated analysis is being used in diverse domains like healthcare, marketing, sports, and security etc [13].

In machine learning, decision making and behavioral analysis is essentially a classification problem, where prediction implies assigning the correct label [14][15]. In [16], Chen et al. proposed an unsupervised deep learning solution to forecast scenarios based on multi-dimensional sensor data representing spatio-

temporal correlations. Their system is data-driven and essentially model-free. In [17], Alonso et al. used machine learning techniques for predicting anomalies in software. They used Random Forest and achieved a validation error of less than 1%. Isupova in [18] did a deep dive into machine learning for behavior analysis and detection of anomalies in videos. The author introduced a novel Bayesian topic models-based approach to model the activities and use them to identify the changes. Arac et al. in [19] explored the brain-to-behavior connection from both neurology and imaging point-of-view and presented a deep learning toolbox using three convolutional neural network architectures for automated analysis.

Statistical machine learning methods are also used for behavior and scenario prediction, however not much attention is placed to video analysis. Umana-Hermosilla et al. in [20] use multinomial logistic regression to predict how individuals and companies perceive the new normal under the COVID-19 pandemic. Westlinder in [21] explored instance-based learning and trained a Support Vector Machine with packet-based features for video traffic classification. In [22], Chakrabarti et al. proposed a fast training method using Sequential Minimal Optimization (SMO) to handle the large training sets and solving the quadratic programming problem that becomes a bottleneck for Support Vector Machines. Bifet et al. in [23] explained the application of Hoeffding Trees in training and classifying data streams.

3 Machine Learning for Treatment Identification using inside-out vision

3.1 Gaze-based Information

Information regarding objects seen in FPV images is important and is used to define the focus of attention. This information is more comprehensive compared to the regular outside-in view, because it does not have occlusion problems. As an example, consider Figure 1 [24], captured via a mounted camera, where we see a dentist performing treatment on a patient. However, the treatment can not be identified correctly because of the occlusion of objects (like tools) by the dentist's hands. Also, the state of the patient's teeth (i.e. symptom) can not be detected from outside-in view, which is another significant parameter in identifying the overall treatment. Now consider Figure 2 [25],[26], in which a similar scene is observed from the first person view of the dentist. Here we can see clearly see the tools used along with the condition of teeth (i.e. symptoms). These inside-out camera observations provide information about the objects seen and help to understand the current scenario.



Fig. 1: Outside-in view: Patient being treated by a dentist [24]. No information about condition of teeth, tools or material used can be extracted from the image due to self-occlusion.



Fig. 2: Inside-out view: Patient being treated by a dentist [25] [26]. Here we can see the tools used (left) and condition of teeth (right)

Each of the dental treatment comprises of a variety of material, symptoms and tools. Conversely, these components allow us to predict the treatment taking place. Consider Figures 3 through 4 that show different dental procedures along with the material, symptoms and tools used in a scenario.



Fig. 3: Dental Treatments (L to R): crown [27], filling [28], imaging [29], sealant [30]



Fig. 4: Material, Symptoms and Tools

The treatments and their respective objects in focus is shown in Table 1. As described above, three parameters are being used to identify the dental treatment and that are: material, symptom and tools.

Table 1: Links between Treatment and Material, Symptoms & Tools for a Dental Setup

Treatment	Material	Symptoms	Tools
Filling	Ceramics, amalgam, glass ionomer, zinc oxide, resin composites	cracked, decomposed, split	air abrasion or laser, condenser, drill, excavator, carver
Crown	resin composites, porcelain fused to metal, metals, zirconia, ceramics, stainless steel	Split, severely worn down, weakened	burs, cement spatula, retraction cord, hand piece, impression trays, probe
Scaling	chlorhexidine gluconate, scalar tips	plaque, calculus	Sonic, ultra sonic
Whitening	carbamide peroxide, hydrogen peroxide	pellicle film	slow handpiece, profy
Sealant	acid solution	premolar, molar	plastic instrument, curing light
Root Canal	resin composites, porcelain fused to metal, metals, zirconia, ceramics, stainless steel, gutta-percha filling	Split, severely worn down, weakened	burs, cement spatula, retraction cord, hand piece, impression trays, probe

3.2 Machine Learning Techniques for Treatment Identification

In this section, we explore a number of machine learning methods for treatment identification in dental setup. We examine Multinomial Logistic Regression, Lazy Instance-based Learning (IBk), Sequential Minimal Optimization (SMO), Hoeffding Tree and Random Tree, and provide a comparative analysis over performance, latency and error metrics. In the remainder of the section, we provide a theoretical and mathematical overview of these techniques.

3.2.1 Multinomial Logistic Regression

Multinomial Logistic Regression is a modeling technique to predict classification of dependent variables based on multiple independent variables - which can be binary or continuous.

In order to calculate the parameters and reduce errors, a number of techniques are used, one of them being ridge estimator [31]. The ridge estimator is used for multinomial logistic regression as follows: Given k classes for n instances each with m attributes, an $m \times (k-1)$ matrix is calculated denoting the parameter matrix B .

The probability for all classes j except for the last class is calculated by Eq. (1)

$$P_j(X_i) = \frac{\exp(X_i B_j)}{\sum_{j=1}^{k-1} \exp(X_i * B_j) + 1} \quad (1)$$

The last class has probability as given in Eq. (2)

$$P_{last}(X_i) = 1 - \sum_{j=1}^{k-1} P_j(X_i) = \frac{1}{\sum_{j=1}^{k-1} \exp(X_i * B_j) + 1} \quad (2)$$

Thus, the multinomial log-likelihood is calculated as in Eq. (3):

$$L = - \sum_{i=1}^n [\sum_{j=1}^{k-1} (Y_{ij} * \ln(P_j(X_i))) + (1 - \sum_{j=1}^{k-1} Y_{ij}) * \ln(1 - \sum_{j=1}^{k-1} P_j(X_i))] + ridge * B^2 \quad (3)$$

A Quasi-Newton Method is used to search for the optimized values of $m * (k - 1)$ variables in order to find the matrix B for which L is minimized. We use it to estimate the local minima since other comprehensive methods can become computationally expensive for each iteration.

3.2.2 Lazy Instance-based Learning (IBk)

Instance-based Learning (IBk) is an implementation of K-nearest neighbors (k-NN) classifier, with the possibility to select an appropriate value of k based on cross-validation and apply distance weighting. A comparative analysis of Bayes and Lazy classification algorithms show [32] that the lazy classifier is more efficient than Bayesian classifier. The simplistic classification algorithm of IBk is given in Algorithm 1:

Algorithm 1 K-Nearest Neighbor Algorithm [32].

- 1: **Class Training**
 - 2: Build the set of training examples D.
 - 3: **Class Classification**
 - 4: Given a query instance x_q to be classified,
 - 5: Let $x_1 \dots x_k$ denote the k instances from D that are nearest to x_q
 - 6: Return
 - 7: $F(x_q) = \arg \max \sum_{i=1}^k \delta(v, f(x_i))$
 - 8: where $(a, b) = 1$, if $a = b$, and $-(a, b) = 0$ else
-

3.2.3 Sequential Minimal Optimization (SMO)

The originally available training method for Support Vector Machine (SVM) were complex and required computationally expensive Quadratic Programming (QP) solvers. Hence, Sequential Minimal Optimization (SMO) algorithm was proposed by Platt in [33] for solving the QP problem arising during SVM training. It is implemented by the LIBSVM tool [34] and has gained much popularity.

Training a Support Vector Machine can be regarded as an optimization problem, which requires solving a quadratic program. Consider, e.g. a binary classification problem [35] with a dataset $(x_1, y_1), \dots, (x_n, y_n)$ where x_i is an

input vector and y_i is the corresponding binary label. A soft-margin SVM is expressed in Eq. (4)

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j K(x_i, x_j \alpha_i \alpha_j) \quad (4)$$

subject to:

$$0 \leq \alpha_i \leq C, \quad \text{for } i = 1, 2 \dots n \quad (5)$$

$$\sum_{i=1}^n y_i \alpha_i = 0 \quad (6)$$

where C is an SVM hyperparameter, $K(x_i, x_j)$ is the kernel function and α_i are Lagrange multipliers.

SMO is an iterative algorithm to solve this optimization problem by breaking it into smaller atomic problems and solving analytically. This allows the constraints on any two Lagrange multipliers α_1 and α_2 to be simplified to:

$$\begin{aligned} 0 &\leq \alpha_1, \alpha_2 \leq C, \\ y_1 \alpha_1 + y_2 \alpha_2 &= k. \end{aligned}$$

This simplified problem can be solved analytically by finding a minimum of a 1-D function over the rest of terms in the equality constraint.

3.2.4 Hoeffding Tree

The Hoeffding Tree [36], also known as Very Fast Decision Tree (VFDT), is a decision tree for classification tasks in data streams. It is an incremental, tree induction algorithm that can learn from massive datasets given that the distribution of samples does not change over time. Hoeffding trees utilize the fact that if an appropriate splitting attribute is chosen then even a small sample set can be sufficient. Mathematically, Hoeffding trees are based on Hoeffding bound [37], which quantifies the number of observations required to estimate the statistical parameters within a prescribed precision. It can be shown that with enough examples (infinite), its output is identical to that of a non-incremental learner.

3.2.5 Random Tree

Random tree, which may also be regarded a decision tree, is essentially the outcome of a stochastic process. It's constructed by considering k randomly chosen attributes at each node without any pruning. Random tree also has an option to allow estimation of class probabilities based on back-fitting.

4 Experiments & Results

4.1 Dataset

For the purpose of experiments we collected our own dataset in the wild. Such a data collection is inherently complicated because of the randomness and unstructured nature of the input. However, it serves as a reliable test set for evaluation of ML models. The data was extracted from real and synthetic videos of dental treatments and used for training and testing the models. A subset of images is shown in the Figure 5.



Fig. 5: Subset of dataset collected in-the-wild for Dental Treatment

Next, we explain the configuration of the machine learning models used for treatment prediction. We used the ML tool WEKA [38] for implementing the models and run the simulations.

4.2 Multinomial Logistic Regression

We used WEKA's logistic regression class with modified ridge estimator to train and run a logistic model. The missing values (if any) were replaced with a missing value's filter and nominal classes were converted to binary. For calculations, we used up to 4 decimal places, batch size of 100 and left the maximum iterations unspecified. The ridge value was initiated to $1.0E-8$.

4.3 Lazy Instance-based Learning (IBk)

The WEKA implementation of Lazy IBk is based on Euclidean distance for nearest neighbor search. We used a batch size of 100, no distance weighting and up to 2 decimal places for calculations. We used a single neighbor for the model ($k = 1$) and trained it without restricting the maximum number of instances in the training pool.

4.4 Sequential Minimal Optimization (SMO)

The SMO function of WEKA is an implementation of the original Platt's proposed algorithm for training an SVM classifier. In this implementation, all missing values are globally replaced and the nominal attributes are transformed into binary. Also, in case of multi-class problems the solution is implemented using pairwise classification i.e. 1-on-1.

For our calculations we use a batch size of 100, a random seed for the cross-validation and up to 2 decimal places. We use the polynomial kernel, the logistic calibrator and normalize training data. The epsilon for round-off error is left to the default value of 1.0E-12.

4.5 Hoeffding Tree

We use the Adaptive Naive Bayes as leaf prediction strategy for implementation of Hoeffding Tree. For this, we set the number of instances that a leaf should observe between split attempts to 200. The split is done according to the info gain split criterion and the allowable error in a split decision is 1.0E-7.

For our calculations we use a batch size of 100 and up to 2 decimal places. The minimum fraction of weight required to go down is set to 0.01 and the hoeffding threshold below which a split will be forced to break ties is 0.05.

4.6 Random Tree

To train a random tree-based model in Weka, we started by specifying the random seed value of 1 for selecting attributes and did not allow unclassified instances for training.

For our calculations, we set the minimum total weight of the instances in a leaf as 1.0, did not apply back-fitting, specified the batch size as 100, and used up to 2 decimal places. In case several attributes look equally good, we did not enable the breaking of ties randomly and allowed the tree to grow to an unlimited depth. The minimum proportion of the variance has to be 0.001 and the number of randomly chosen attributes is calculated by: $\text{int}(\log_2(\text{No. of predictors}) + 1)$.

We present the results in Table 2. We note that objects in gaze are a strong predictor for recognizing treatment even if only one of the three criteria i.e.

material, symptom and tools is taken into account. Moreover, by combining multiple parameters improves the overall accuracy.

Table 2: Treatment Identification Results

Method	Material		Symptoms		Tools		Combined	
	Precision %	Recall %						
Logistic	68.7	68.4	97.5	97.1	94.1	89.8	88.7	78.6
Lazy IBK	80.4	73.7	81.7	61.7	90.4	76.3	1	1
SMO	81.9	68.4	88.8	85.3	87.7	74.6	95	83.3
Tree Ho- effding	81.3	71.1	97.5	97.1	90.4	76.3	92.9	85.7
Random Tree	84.9	55.3	90.5	79.4	80.4	54.2	94	61.9

With extracted object information in focus, the ML models were regenerated and trained. As expected, in most cases the precision and recall were improved when considering the combined information compared to when the parameters were used individually. It is surprising to note that for Multinomial Logistic Regression and Tree Hoeffding methods, the best precision and recall were achieved when Symptoms parameter was used independently.

We also looked at the training and testing times (in sec) of each of these models and present them in Table 3. Here, times smaller than 0.01 sec were automatically rounded off to zero. We notice that the combined models were the slowest to build and test, hence have a disadvantage for computationally intensive applications or for huge datasets. Out of these the SMO took longest to build and Hoeffding Tree took the longest to test.

Table 3: Time (in sec) required to build and test the models

Techniques	Material		Symptoms		Tools		Combined	
	Build	Test	Build	Test	Build	Test	Build	Test
Logistic	0.01	0	0.01	0	0	0	0.02	0.03
Lazy IBK	0	0.1	0	0	0	0	0	0.02
SMO	0.06	0	0.06	0	0.06	0	0.14	0.03
Hoeffding Tree	0	0	0	0	0	0	0.01	0.06
Random Tree	0	0	0	0	0	0	0	0.02

Finally, we compare the Root Mean Squared Error (RMS) of the different models and found that in most of the cases Symptoms parameter returned the minimum RMS error.

Table 4: Root Mean Squared Error

Techniques	Material	Symptoms	Tool	Combined
Logistic	0.3088	0.1064	0.1743	0.2673
Lazy IBK	0.3059	0.2564	0.1989	0.2113
SMO	0.3397	0.3165	0.3227	0.3277
Hoeffding Tree	0.318	0.0762	0.2026	0.2182
Random Tree	0.389	0.2039	0.3906	0.3563

5 Conclusion

In this paper, we used the dental setup and presented treatment identification system with the objects in focus that includes materials, symptoms and tools. We have trained and tested various machine learning models including Multinomial Logistic Regression, Lazy Instance-based Learning (IBk), Sequential Minimal Optimization (SMO), Hoeffding Tree and Random Tree. Combining the observations w.r.t. performance, speed and error convergence, we conclude that combining multiple parameters result in better performance in some cases, however it results in low error convergence and slow speed. Also, not all machine learning techniques give an improved performance for combined approach. Rather, the best trade-off is received when we use symptoms as a stand-alone parameter. This shows that in machine learning applications in general and in medical/dental applications in particular, adding more parameters does not always lead to improved results. Rather it depends on the ML tool used, the parameters considered and the data given as input.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Land M.F., Vision, eye movements, and natural behavior, *Visual Neuroscience*, vol. 26, no. 1, pp. 51 – 62, (2009).
2. Jacob R.J.K. and Karn K.S., Eye Tracking in Human Computer Interaction and Usability Research: Ready to Deliver the Promises, *J. The Mind’s Eye: Cognitive and Applied Aspects of Eye Movement Research*, pp. 573 – 605, (2003).
3. Yarbus A.L., *Eye Movements and Vision*, Plenum., (1967).
4. Hayhoe M. and Ballard D., Eye movements in natural behavior, *J. Trends in Cognitive Sciences*, vol. 9, no. 4, pp. 188 – 194, 03 March (2005).
5. Zelinsky G.J., Rao R.P.N., Hayhoe M.M., and Ballard D.H., Eye Movements Reveal the Spatiotemporal Dynamics of Visual Search, *A journal of the association for Psychological Science*, vol. 8, no. 6, pp. 448 – 453, (1997).
6. Toet A., Gaze directed displays as an enabling technology for attention aware systems, *Journal in Computers in Human Behavior*, vol. 22, no. 4, pp. 615 – 647, (2006).
7. Henderson J. M., & Hollingworth A. High-level scene perception. *Annual Review of Psychology*, 50, 243–271. (1999).

8. Rayner K., Smith T. J., Malcolm G. L., & Henderson J. M., Eye movements and visual encoding during scene perception. *Psychological Science*, vol. 20 issue 1, 6 – 10. (2009).
9. Zhou B., Lapedriza A., Xiao J., Torralba A., & Oliva, A. Learning deep features for scene recognition using places database. In Z. Ghahramani M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems* Vol. 27, pp. 487–495. Red Hook: Curran Associates Inc. (2014).
10. Kang H., Alexei A.E., Herbert, M., and Kanade T., Image Matching in Large Scale Indoor Environment, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshop on Egocentric Vision*, (2009).
11. Sun L., Klank U., and Beetz M., EYEWATCHME 3D Hand and object tracking for inside out activity analysis, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 9 – 16, (20 - 25 June 2009).
12. Furnari A. , Farinella G. M., What Would You Expect? Anticipating Egocentric Actions with Rolling-Unrolling LSTMs and Modality Attention, *International Conference on Computer Vision*, (2019).
13. Enrique Garcia Ceja, *Behavior Analysis with Machine Learning Using R*, (2021) Book Link, <https://enriquegit.github.io/behavior-free/citing-this-book.html>
14. *Machine Learning Applications* (2018) Link: <http://intelligentonlinetools.com/blog/2018/08/11/applied-machine-learning-classification-for-decision-making>
15. Kashyap P., *Machine Learning for Decision Makers - Cognitive Computing Fundamentals for Better Decision Making* (2017) doi: <https://doi.org/10.1007/978-1-4842-2988-0>
16. Chen Y., Wang X. and Zhang B. An Unsupervised Deep Learning Approach for Scenario Forecasts (2018) [Online available at]: <https://arxiv.org/pdf/1711.02247.pdf>
17. Alonso J., Belanche L. and Avresky D.R. Predicting Software Anomalies using Machine Learning Techniques. Online available at: <https://core.ac.uk/download/pdf/41767353.pdf>
18. Isupova O., *Machine Learning Methods for Behaviour Analysis and Anomaly Detection in Video*, Springer International Publishing, (2018) doi: 10.1007/978-3-319-75508-3
19. Arac A, Zhao P, Dobkin BH, Carmichael ST and Golshani P., DeepBehavior: A Deep Learning Toolbox for Automated Analysis of Animal and Human Behavior Imaging Data. *Front. Syst. Neurosci.* 13:20. (2019) doi: 10.3389/fnsys.2019.00020
20. Umaña-Hermosilla B., Fuente-Mella de L., Elórtegui-Gómez C. and Fonseca-Fuentes M., Multinomial Logistic Regression to Estimate and Predict the Perceptions of Individuals and Companies in the Face of the COVID-19 Pandemic in the Ñuble Region, Chile, *Sustainability*, vol 12 issue 22, Article No. 9553 (2020), doi: 10.3390/su12229553
21. Westlinder S., Video Traffic Classification A Machine Learning approach with Packet Based Features using Support Vector Machine, *Digitala Vetenskapliga Arkivet, Computer Sciences*, pp 88(2016)
22. Kumar B., Sinha A., Chakrabarti S. and Khandelwal A., Sequential Minimal Optimization for One-class Support Vector Machine, *International Conference on Data Mining and Software Engineering* (2020)
23. Bifet A., Gavaldà R., Holmes G., Pfahringer B. *MACHINE LEARNING FOR DATA STREAMS with Practical Examples in MOA*, The MIT Press, (2018) doi: 10.7551/mitpress/10654.001.0001
24. [Last seen on: 31-Jan-2020] <https://en.wikipedia.org/wiki/Dentist>
25. [Last seen on: 31-Jan-2020] <https://www.indushealthplus.com/oral-health-care.html>
26. [Last seen on: 31-Jan-2020] <https://www.canstockphoto.com/images-photos/gold-teeth.html>
27. [Last seen on: 23-Feb-2020] <http://www.ismiledental.com.sg/dental-implant/>
28. [Last seen on: 23-Feb-2020] <https://www.isiltidisler.com/beyaz-kompozit-dolgu?lang=en>
29. [Last seen on: 23-Feb-2020] <https://www.qandm.com.sg/Q&M-Dental-Group-Singapore-Services-General-Dental-Treatment>
30. [Last seen on: 23-Feb-2020] <http://www.dudhatdentalclinic.com/service.php>
31. Cessie S.Le, Houwelingen, J.C.van, Ridge Estimators in Logistic Regression. *Applied Statistics*. vol. 41, issue 1, pp. 191-201. (1992).

32. Vijayarani S. and Muthulakshmi M., Comparative Analysis of Bayes and Lazy Classification Algorithms, *International Journal of Advanced Research in Computer and Communication Engineering* Vol. 2, Issue 8, (August 2013)
33. J. Platt: Fast Training of Support Vector Machines using Sequential Minimal Optimization. In B. Schoelkopf and C. Burges and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, (1998).
34. Chang, Chih-Chung; Lin Chih-Jen, LIBSVM: A library for support vector machines, *ACM Transactions on Intelligent Systems and Technology*, vol. 2, issue 3 (2011).
35. [Last seen on: 17-May-2021]
https://en.wikipedia.org/wiki/Sequential_minimal_optimization#cite_note-2
36. Hulten G., Spencer L., Domingos P., Mining time-changing data streams. *Proceeding of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 97-106, (2001).
37. [Last seen on: 17-May-2021] https://en.wikipedia.org/wiki/Hoeffding%27s_inequality
38. [Last seen on: 17-May-2021] <https://www.cs.waikato.ac.nz/ml/weka/>

Figures



Figure 1

Outside-in view: Patient being treated by a dentist [24]. No information about condition of teeth, tools or material used can be extracted from the image due to self-occlusion.



Figure 2

Inside-out view: Patient being teated by a dentist [25] [26]. Here we can see the tools used (left) and condition of teeth (right)



Figure 3

Dental Treatments (L to R): crown [27], filling [28], imaging [29], sealant [30]



Figure 4

Material, Symptoms and Tools



Figure 5

Subset of dataset collected in-the-wild for Dental Treatment