

# Labeling System with EEG, EMG, and IMU for Visual Training of Autonomous Vehicles

*Araştırma Makalesi/Research Article*

 Ahmet Çağdaş SEÇKİN

Elektronik ve Otomasyon Bölümü, Teknik Bilimler Meslek Yüksekokulu, Uşak Üniversitesi, Uşak, Turkey  
[cagdas.seckin@usak.edu.tr](mailto:cagdas.seckin@usak.edu.tr)

(Geliş/Received:21.03.2019; Kabul/Accepted:07.08.2019)

DOI: 10.17671/gazibtd.542662

**Abstract**— Autonomous vehicles are tools that make decisions and take decisions by perceiving their environment. Today, autonomous vehicles are also used in traffic in some countries. Various types of cameras, laser radars (LIDAR), sonar distance sensors, etc. are used for environmental detection in autonomous vehicles. After the environment is perceived, the collected data is taught to the vehicle with the help of machine learning methods and the vehicle reaches the target by following the traffic rules. At the point of traffic rules, the biggest task belongs to image-based systems. However, ideal traffic conditions and environmental conditions are not always provided. It is important to identify situations that may present a danger to autonomous vehicles. When the literature is examined, no visual data set or a scientific study with dangerous labeling has been found. In this study, it is aimed to design a data collection and labeling system to overcome this gap in the literature. In the system designed for the purpose, a system which automatically creates a video label from the physiological data of the driver (EEG ve EMG) and the inertia change data during human driving is designed. For this reason, firstly, the sensor signals were collected by experiments. In the time and frequency field, attributes were extracted by using the non-overlapping sliding window with 0.33 sec length. The input variables in the data set were reduced by PCA and classified by DT, RF and K-NN algorithms. According to the preliminary study findings, the K-NN method was the most successful algorithm among the algorithms tested with 0.922 accuracy.

**Keywords**— EEG, EMG, IMU, autonomous vehicles, human-machine interaction, machine learning

## Otonom Araçların Görsel Eğitimi için EEG, EMG ve IMU ile Etiketleme Sistemi

**Özet**— Otonom araçlar, çevrelerini algılayarak kararlar alan ve bu kararlar ile hareket eden araçlardır. Günümüzde otonom araçlar bazı ülkelerde trafikte de kullanılmaktadır. Otonom araçlarda çevre algılama için çeşitli kameralar, lazer radarlar (LIDAR), sonar sensörler vb. pek çok sensör kullanılmaktadır. Çevre algılandıktan sonra toplanan veri makine öğrenmesi yöntemleri yardımıyla araca öğretilmekte ve araç trafik kurallarına uyarak hedefe ulaşmaktadır. Trafik kuralları noktasında en büyük görev görüntü tabanlı sistemlere düşmektedir. Ancak ideal trafik koşulları ve çevre şartları her zaman sağlanamamaktadır. Bu nedenle otonom araçlar için tehlike oluşturabilecek durumların tespiti önem arz etmektedir. Literatür incelendiğinde tehlikeli durumların etiketli bulunduğu görsel bir veri seti veya ilgili bir bilimsel çalışmaya rastlanmamıştır. Bu çalışmada literatürdeki açığı gidermek için bir veri toplama ve etiketleme sistemi tasarlanması amaçlanmıştır. Amaç doğrultusunda tasarlanan sistemde insan sürüşü esnasında sürücünün fizyolojik verisi (EEG ve EMG) ve eylemsizlik değişim verilerinden otomatik olarak video etiketi oluşturan bir sistem tasarlanmıştır. Bunun için öncelikle deneyler ile sensör sinyalleri toplanmıştır. Toplanan sinyallerden 0.33 sn uzunluğunda üst üste binmeyen kayan pencere kullanılarak zaman ve frekans alanında öznitelikler çıkarılmıştır. Elde edilen veri setindeki giriş değişkenleri Temel Bileşen Analizi (PCA) ile indirgenmiş ve Karar Ağacı (DT), Rastgele Ağaç (RF) ve K En Yakın Komşular (K-NN) algoritmaları ile sınıflandırma işlemine tutulmuştur. Bulgulara göre K-NN yönteminin 0.922 doğrulukla tehlikeli, tehlikesiz durumları ayırt ederek denenen algoritmalar arasında en başarılı algoritma olduğu tespit edilmiştir.

**Anahtar Kelimeler**— EEG, EMG, IMU, otonom araçlar, insan makine etkileşimi, makine öğrenmesi

## 1. INTRODUCTION

Autonomous vehicles are tools that take decisions by sensing their environment and act with these decisions [1]. Many sensors like cameras, laser radars (LIDAR), sonar sensors, etc. are used for motion planning according to environmental conditions in autonomous vehicles. [1], [2]. After sensing the environment, the collected data is taught to the vehicle with the help of artificial learning and the vehicle learns to reach the target by following the traffic rules.

Autonomous vehicles learn to comply with the traffic rules the most important task falls to visual systems. Traffic signs are mostly recognized through the visual object and sign recognition. Because systems like LIDAR and GPS are very efficient in route planning and avoidance of obstacles however traffic signal perception, lane detection and infrastructure problems (bump, rail, pit, etc.) cannot be recognized very efficiently. Also, ideal traffic conditions and environmental conditions cannot be provided all the time and all the places. According to researches, most car accidents are caused by human factor[4]–[6]. Although many research institutions and vehicle manufacturers recently have focused on the commercialization of autonomous driving systems, the aforementioned situations are a major problem in terms of commercialization and dissemination.

In addition, the decision-making mechanisms of autonomous instruments in critical situations in terms of legal and ethical rules have been examined in various studies. [7]–[10]. The ethical decision-making of artificial intelligence in its hazard or critical situations is based on data taught to artificial intelligence. This process is based on the principle of introducing objects to artificial intelligence and giving priority to object-position relationships [11]–[15]. The priority recognition process that causes ethical debates is not simple enough to be reduced to an equation.

When the studies on motion planning, safety, and reliability of autonomous vehicles are examined, main applications are obstacle avoidance-collision avoidance[16], [17], odometry [2], [18], [3], lane tracking [19], [20]. Today, many types of research on autonomous vehicle development and the data sets obtained as a result of these studies are shared as open source[21]–[23]. The most comprehensive data set consisting of LIDAR, IMU, GPS and camera (single and multiple) navigation data that can be used in autonomous vehicles is the KITTI data set prepared by Karlsruhe Institute of Technology [24], [25]. However, when these studies were examined, a data set with dangerous or critical conditions was not available.

In this study, it is aimed to design a system that makes automatic labeling by taking advantage of the physiological data of the driver and the physical effects during driving in order to fill the gap in the literature. In this context, a system for the simultaneous collection of

Inertia Measurement Unit (IMU) Sensor, Electroencephalography (EEG), Electromyography (EMG) and driving video data has been established. Some of the videos collected with this system are monitored and the situations that may cause danger are marked in the time stream. The patterns in which the IMU, EEG and EMG signals formed at the marked time intervals occur are considered as dangerous state patterns. The patterns were converted into a standard labeling model with machine learning. With this model, it is aimed to provide convenience, labor saving and time-saving by making automatic labeling of dangerous situation zones in the videos. In the following parts of the article, the materials, designed system, experiments, findings, and conclusion are presented.

## 2. MATERIAL AND METHOD

System design and connections are given in Figure-1. The system basically consists of a computer, IMU, EEG, EMG sensors and one camera. The EMG and IMU sensors are combined with a microcontroller as shown in Figure-2 and connected to a wrist bandage by means of a patch for easy wear. The camera records 30 frames per second. Simultaneous recording is performed because the signals obtained for each frame will be examined. In the signal analysis, the sliding window length is 0.033 sec and the windows will be processed without overlapping.

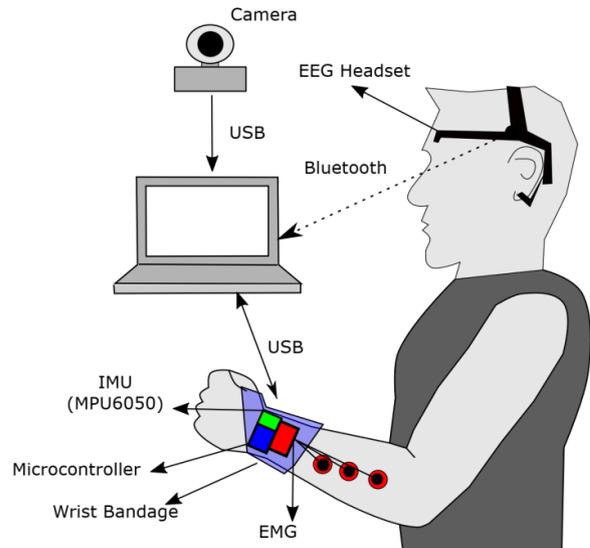


Figure 1. System Design

The IMU sensor is a microelectromechanical device used to measure the 3-axis acceleration and the angular rotation of a moving object in space. MPU6050 sensor is used as an IMU sensor. The MPU6050 sensor is connected to the microcontroller via the Inter-Integrated Circuit (I2C) connection and 100 samples are taken per second. The maximum limit for this sensor was selected as 4G because it was observed that even the F1 vehicles, which could be considered the fastest road vehicles, reached a maximum of three times the gravity [26]. Only 3 axis acceleration data of the MPU6050 sensor were used in the study. The mean, standard deviation, maximum, minimum and the

Root Mean Square (RMS) values of these data were used as features.



Figure 2. EMG and IMU Wrist Bandage

EEG sensors have long been used in the medical field to invasively study neural activity in the brain. Conventional EEG devices are designed to take measurements from multiple points in the 10-20 placement method [27]. The EEG device to be used in this study is the MindWave Mobile device of Neurosky, which is designed as an easily wearable headset. It produces output as a single channel by measuring the potential difference between A1 and Fp1 in the 10-20 method. The MindWave device can be used to measure emotions such as attention, meditation, blinking and mental activity[28]. In addition, this device has been used to detect brain activity in many areas such as education[29], [30], human-machine interaction[31], [32] and robotics[33]. In this study, 512 samples are taken from the EEG sensor per second. Time and frequency domain data of EEG data were used for machine learning. Mean, standard deviation, maximum and RMS values were used as features in the time domain. The frequency range of the EEG signals is 0-100 Hz. reduction own

internal totals were used as 5 separate attributes. The frequency domain data is automatically supplied by the device. However, in order to reduce the number of attributes, only the internal totals of main frequency bands which are delta (0-4 Hz), theta (4-7 Hz), alpha (8-12 Hz), beta (12-30 Hz) and gamma (30-100 Hz) are used as 5 different features.

EMG sensors are used to detect electrical field changes caused by muscle activity. The EMG sensor to be used in the study is the AD8232 module. This sensor generates signals about muscle activity with 3-point measurement and has been used in many studies [34]–[36]. The signal is received from the EMG sensor in the range of 10-1000Hz. The mean, standard deviation, maximum, minimum and RMS values were used as features in the time domain from the EMG sensor. In the frequency domain, only peak frequency and amplitude are used as features.

### 2.1. Machine Learning

The data set consists of 15 from the IMU sensor, 10 from the EEG sensor and 7 from the EMG sensor with a total of 32 features which are mentioned in the previous section. These variables were reduced to 10 components by Principal Component Analysis (PCA)[37]. The method for machine learning is shown in Figure-3. Accordingly, the data set with 10 features obtained after PCA was divided into 70% training and 30% test data. The training data was applied to the Decision Tree (DT), Random Forest (RF) and K Nearest Neighbors (K-NN) algorithms and the best performance metric based on the classification results were chosen as the label generation model.

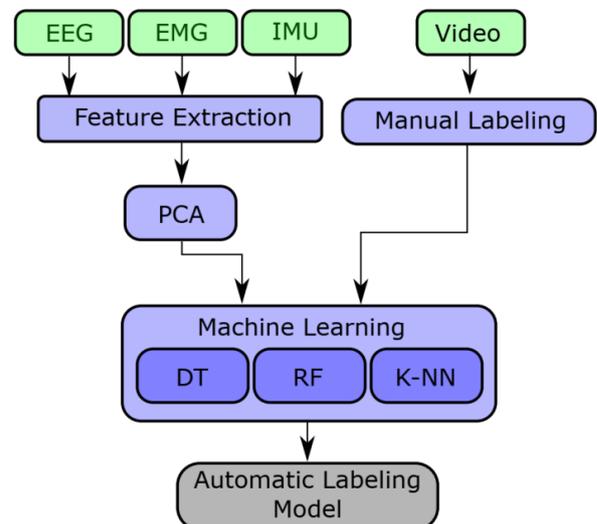


Figure 3. Machine Learning Flowchart

The DT algorithm is an algorithm frequently used in statistical learning and data mining. DT is an algorithm used for both classification and regression in supervised learning with different decision rules than simple decision[38]. The decision tree forms a tree structure classification or regression models. It divides a dataset into smaller subgroups and is also incrementally developed.

Decision trees can use both categorical and numerical data[39]. The DT algorithm has three basic steps. The first step is placed as the first (root) node. In the first step, the most significant feature is placed as the first (root) node. In the second step, the data sets are divided into subsets according to this node. The partition operation must be made to contain the same value of data for a feature of each subset. In the third step, the first and second steps are repeated until the last (leaf) nodes in all branches are found.

The RF algorithm is designed as a forest of many DTs. Each decision tree in the forest is formed by selecting the sample from the original data set with the bootstrap technique and selecting the random number of all variables in each decision node. The RF algorithm consists of four fundamental steps. First of all, randomly select  $n$  features from total  $m$  features. For the second step, among the  $n$  features, calculate the node  $d$  using the best split point. In the third step check whether the number of final (leaf) nodes reached to target if it is not, go to step one otherwise go to next step. For the last step build forest by repeating steps one to three for  $n$  (number of trees in the forest) times [40]–[42].

The K Nearest Neighbors (KNN) algorithm is a learning algorithm that works according to the values of the nearest  $k$ -neighbor. The KNN algorithm is a non-parametric method for classification and regression [57]. It was first applied to the classification of news articles [58]. When performing learning with the KNN algorithm, firstly the distance of each data to the other is calculated in the data set examined. This length calculation is done with Euclidian, Manhattan or Hamming distance function. Then for each data mean of nearest  $K$  neighbors is calculated. The  $K$  value is the only hyperparameter of the KNN algorithm. When deciding  $K$  value, If the  $K$  is too low, then the borders are going to be flickering and overfit situation occurs, whereas if the  $K$  value is too high, the separation borders going to be smoother and underfit situation occurs. The disadvantage of the KNN algorithm is the distance calculation process because it increases the processing load as the number of data increases.

There are a number of model evaluation techniques but some of the most well-known are percentage split and cross-validation. In the evaluation processes, it is essential to use a train and a test data set. Percentage split is the most basic method. In this method, all the data are split as train and test by manually. Training data set are used for the learning process and the test data set is used for performance evaluation. However, evaluation results may not be reliable because of the reasons such as not having the same distribution when selecting the train and test data in the data set, uneven distribution of outliers and so on. Therefore, Cross Validation method has been developed. In this method, train and test data are integrated and turned into a single data set. All data are divided into  $K$  equal-sized sub sets. The  $K$  value which is also called as fold number is determined by the user. Then, learning and testing are performed for each of the  $K$  sub-sets; here one of the subsets will be test, and the other will be train. As a

result, performance metrics are obtained for each sub-set. The average of the performance metrics is considered as the performance metric of the K-Fold Cross-Validation. K-Fold Cross Validation method is known to produce more reliable results than other methods. However, since learning and testing for each subset are performed separately for all subsets, the total time spent longer than the other methods[43].

The main criteria used for performance evaluation and model selection are called metrics. Classification performance metrics are obtained through Confusion Matrix which is shown in Figure-6 as outlined in Table-7. The TP value in a two-class Confusion Matrix is the number of predictions where the predicted value is 1 (true) when the actual value is also 1 (true). The TN value is the number of predictions where the predicted value is 0 (false) when the actual value is also 0 (false). The FP value is the number of predictions where the predicted value is 1 (true) when the actual value is 0 (false). The FN value is the number of predictions where the predicted value is 0 (false) when the actual value is 1 (true)[43], [44].

		Predicted	
		Positives	Negatives
Actual	Positives	True Positives (TP)	False Negatives (FN)
	Negatives	False Positives (FP)	True Negatives (TN)

Figure 4. Confusion matrix

Table 1. Performance metrics of classification

Metric	Equation
Accuracy	$A = \frac{TN + TP}{TP + TN + FP + FN}$
Precision	$P = \frac{TP}{TP + FP}$
Recall	$R = \frac{TP}{TP + FN}$
F1 Score	$F1 = \frac{P + R}{2}$

### 3. EXPERIMENTS AND RESULTS

For the data collection, the experimental setup shown in Figure-5 was established. Five volunteers were asked to drive at least 15 minutes for the experiments and the data were recorded. By monitoring video and signals, deceleration, caution, danger, traffic light, pedestrian crossing, and intersections are labeled with a value of 1 because they can be an example of dangerous situations. Other conditions are labeled 0. As a result, a total of 147506 square images were obtained. However, it is exhausting to mark all of these frames manually. For this reason, a total of 300 frames (150 frames as danger and 150 frames as safe.) were labeled. 210 of these data were used for testing and 90 for the test. Labels and corresponding attributes are applied to machine learning methods. Test classification performance of DT, RF and K-NN algorithms are given in Table-2. The confusion matrices of

the DT, RF, and K-NN algorithms are shown in Figure-3, Figure-4, and Figure-5 respectively. Examples of dangerous and safe conditions obtained during the experiments are presented in Figure-6 and Figure-7, respectively.

Table 2. Comparison of classification performance

Algorithm	Accuracy	Precision	Recall	F1
DT	0.844	0.811	0.915	0.860
RF	0.867	0.863	0.898	0.880
K-NN	0.922	0.918	0.938	0.927

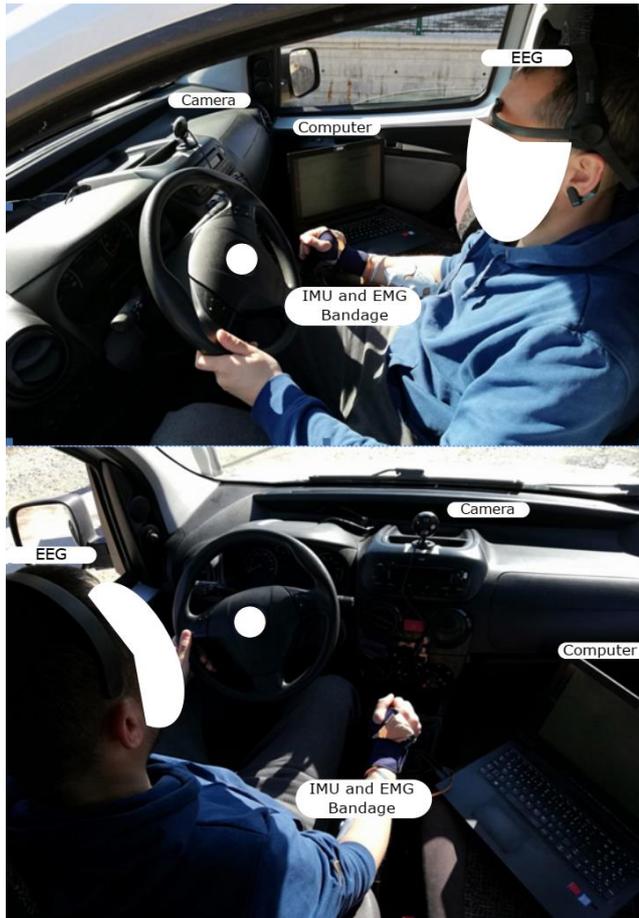


Figure 5. Experimental setup

Table 3. Confusion matrix of DT

		Prediction		Total
		1	0	
Actual	1	43	10	53
	0	4	33	37
Total		47	43	90

Table 4. Confusion matrix of RF

		Prediction		Total
		1	0	
Actual	1	44	7	51
	0	5	34	39
Total		49	41	90

Table 5. Confusion matrix of K-NN

		Prediction		Total
		1	0	
Actual	1	45	4	49
	0	3	38	41
Total		48	42	90



Figure 6. Frame Labeled as Dangerous



Figure 7. Frame labeled as safe

#### 4. CONCLUSION

The aim of the study is to create an automatic labeling system with physiological data to prevent the loss of time and labor during manual labeling of dangerous situations in videos collected during autonomous vehicle use. In line with the objective, a system for the IMU, EEG, EMG, and driving video data were established simultaneously. Some of the videos collected with this system have been watched and potentially dangerous situations are marked in the time stream. Patterns in which IMU, EEG and EMG signals formed at specified time intervals occur were considered as dangerous state patterns. The system designed to fill the gap in the literature regarding the determination of situations that may be dangerous for autonomous vehicles has been implemented successfully. The input variables in the data set were reduced by PCA and classified by DT, RF and K-NN algorithms. According to the preliminary study findings, the K-NN method was the most successful algorithm among the algorithms tested with 0.922 accuracy. With the designed system, it is important to carry out the labeling of dangerous situations successfully, as it

will also provide infrastructure for applications such as summarizing and driving rating systems from driving videos. With this model, convenience, labor saving and time-saving are provided by automatic labeling of danger areas in videos. We will try to reduce the physiological sensors used in future studies.

## REFERENCES

- [1] B. Siciliano, O. Khatib, **Springer Handbook of Robotics**, Springer, 2016.
- [2] D. Nistér, O. Naroditsky, J. Bergen, “Visual Odometry for Ground Vehicle Applications”, *Journal of Field Robotics*, 23(1), 3–20, 2006.
- [3] M. O. Aqel, M. H. Marhaban, M. I. Sariapan, N. B. Ismail, “Review of visual odometry: types, approaches, challenges, and applications”, *SpringerPlus*, 5(1), 1897, 2016.
- [4] H. Lum, J. A. Reagan, “Interactive highway safety design model: accident predictive module”, *Public Roads*, 58(3), 1995.
- [5] T. Bliss, **Implementing the recommendations of the world report on road traffic injury prevention**, 2004.
- [6] M. Peden *et al.*, **World report on road traffic injury prevention**, World Health Organization Geneva, 2004.
- [7] N. J. Goodall, “Can you program ethics into a self-driving car?”, *IEEE Spectrum*, 53(6), 28–58, 2016.
- [8] D. Birnbacher, W. Birnbacher, “Fully autonomous driving: Where technology and ethics meet”, *IEEE Intelligent Systems*, 32(5), 3–4, 2017.
- [9] N. A. Greenblatt, “Self-driving cars and the law”, *IEEE spectrum*, 53(2), 46–51, 2016.
- [10] J. Fleetwood, “Public health, ethics, and autonomous vehicles”, *American journal of public health*, 107(4), 532–537, 2017.
- [11] K. Sadeghi, A. Banerjee, J. Sohankar, S. K. Gupta, “Safedrive: An autonomous driver safety application in aware cities”, **Pervasive Computing and Communication Workshops (PerCom Workshops), 2016 IEEE International Conference on**, 1–6, 2016.
- [12] C. Richter, N. Roy, “Safe visual navigation via deep learning and novelty detection”, 2017.
- [13] D. Gruyer, V. Magnier, K. Hamdi, L. Claussmann, O. Orfila, A. Rakotonirainy, “Perception, information processing and modeling: Critical stages for autonomous driving applications”, *Annual Reviews in Control*, 44, 323–341, 2017.
- [14] N. Smolyanskiy, A. Kamenev, J. D. Smith, S. T. Birchfield, “Performing autonomous path navigation using deep neural networks”, Oct-2018.
- [15] L. Chi, Y. Mu, “Deep steering: Learning end-to-end driving model from spatial and temporal visual cues”, *arXiv preprint arXiv:1708.03798*, 2017.
- [16] A. Bemporad, A. De Luca, G. Oriolo, “Local incremental planning for a car-like robot navigating among obstacles”, **Proceedings of IEEE International Conference on Robotics and Automation**, 2, 1205–1211, 1996.
- [17] J. Liu, P. Jayakumar, J. L. Stein, T. Ersal, “A study on model fidelity for model predictive control-based obstacle avoidance in high-speed autonomous ground vehicles”, *Vehicle System Dynamics*, 54(11), 1629–1650, 2016.
- [18] N. Nourani-Vatani, J. Roberts, M. V. Srinivasan, “Practical visual odometry for car-like vehicles”, in **2009 IEEE International Conference on Robotics and Automation**, 3551–3557, 2009.
- [19] F. You, R. Zhang, G. Lie, H. Wang, H. Wen, J. Xu, “Trajectory planning and tracking control for autonomous lane change maneuver based on the cooperative vehicle infrastructure system”, *Expert Systems with Applications*, 42(14), 5932–5946, 2015.
- [20] J. Sattar, J. Mo, “SafeDrive: A Robust Lane Tracking System for Autonomous and Assisted Driving Under Limited Visibility”, *arXiv preprint arXiv:1701.08449*, 2017.
- [21] M. Cordts *et al.*, “The cityscapes dataset for semantic urban scene understanding”, **Proceedings of the IEEE conference on computer vision and pattern recognition**, 3213–3223, 2016.
- [22] W. Maddern, G. Pascoe, C. Linegar, P. Newman, “1 year, 1000 km: The Oxford RobotCar dataset”, *The International Journal of Robotics Research*, 36(1), 3–15, 2017.
- [23] F. Yu *et al.*, “BDD100K: A diverse driving video database with scalable annotation tooling”, *arXiv preprint arXiv:1805.04687*, 2018.
- [24] A. Geiger, P. Lenz, R. Urtasun, “Are we ready for autonomous driving? the kitti vision benchmark suite”, **IEEE Conference on Computer Vision and Pattern Recognition**, 3354–3361, 2012.
- [25] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, “Vision meets robotics: The KITTI dataset”, *The International Journal of Robotics Research*, 32(11), 1231–1237, 2013.
- [26] E. S. Watkins, “The physiology and pathology of formula one Grand Prix motor racing”, *Clin Neurosurg*, 53, 145–152, 2006.
- [27] M. Quigg, M. Quigg, **EEG pearls**, Mosby Elsevier, 2006.
- [28] Internet: “EEG Algorithms | NeuroSky” . <http://neurosky.com/biosensors/eeg-sensor/algorithms/>
- [29] A. Sezer, Y. İnel, A. Ç. Seçkin, U. Uluçınar, “An Investigation of University Students’ Attention Levels in Real Classroom Settings with NeuroSky’s MindWave Mobile (EEG) Device”, **International Educational Technology Conference 2015**, 88–101, 2015.
- [30] A. Sezer, Y. İnel, A. Ç. Seçkin, U. Uluçınar, “The Relationship between Attention Levels and Class Participation of First-Year Students in Classroom Teaching Departments”, *International Journal of Instruction*, 10(2), 55–68, 2017.
- [31] F. Bozkurt, A. Ç. Seçkin, A. Coşkun, “Integration of IMU Sensor on Low-Cost EEG and Design of Cursor Control System with ANFIS”, *International Journal of Engineering Trends and Technology*, 54(3), 162–169, 2017.
- [32] K. Patel, H. Shah, M. Dcosta, D. Shastri, “Evaluating NeuroSky’s Single-Channel EEG Sensor for Drowsiness Detection”, *HCI International 2017 – Posters’ Extended Abstracts*, 243–250, 2017.
- [33] P. D. Girase, M. P. Deshmukh, “Mindwave Device Wheelchair Control”, 2013.

- [34] B. Champaty, P. Dubey, S. Sahoo, S. S. Ray, K. Pal, A. Anis, "Development of wireless EMG control system for rehabilitation devices", **Annual International Conference on Emerging Research Areas: Magnetism, Machines and Drives (AICERA/iCMMD)**, 1–4, 2014.
- [35] M. A. Ahamed, M. A.-U. Ahad, M. H. A. Sohag, M. Ahmad, "Development of low cost wireless biosignal acquisition system for ECG EMG and EOG", **2nd International Conference on Electrical Information and Communication Technologies (EICT)**, 195–199, 2015.
- [36] N. Mulayim, S. Cıkcandır, "Low-Cost Real-Time Electromyography (EMG) Data Acquisition Experimental Setup for Biomedical Technologies Education", 7, 2017.
- [37] I. Jolliffe, **Principal component analysis**, Springer, 2011.
- [38] J. R. Quinlan, "Induction of decision trees", *Machine learning*, 1(1), 81–106, 1986.
- [39] J. R. Quinlan, "Simplifying decision trees", *International journal of man-machine studies*, 27(3), 221–234, 1987.
- [40] L. Breiman, "Random forests", *Machine learning*, 45(1), 5–32, 2001.
- [41] A. Liaw, M. Wiener, "Classification and regression by randomForest", *R news*, 2(3), 18–22, 2002.
- [42] M. Akman, Y. Genç, H. Ankaralı, "Random forests yöntemi ve sağlık alanında bir uygulama", *Türkiye Klinikleri Journal of Biostatistics*, 3(1), 36–48, 2011.
- [43] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression", *The American Statistician*, 46(3), 175–185, 1992.
- [44] B. Masand, G. Linoff, D. Waltz, "Classifying news stories using memory based reasoning", **15th annual international ACM SIGIR conference on Research and development in information retrieval**, 59–65, 1992.
- [45] E. Alpaydin, **Introduction to machine learning**. MIT press, 2009.