JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Abnormal Activity Detection

Tejas Pethkar[1], Megha Pallewar[2], Lakhan Musale[3], Amit Anand[4]

Department of E&TC, SKNCOE, SPPU, Pune

Abstract— **Abnormal activity will lead to the uncommon changes in the crowd behavior. In other words, the crowd motion changes conforms to certain rules for valid behaviors, while for abnormal events the motion changes are uncontrolled. The motion-changed rules to detect and localize abnormal behavior in crowd videos. Specifically, we first generate the motion patterns based on the descriptor of collectiveness. Then each frame pair is represented as a transfer matrix whose elements are the difference of a set of motion patterns. Thereafter, the motion-changed rules are constructed in the transformation space using bag-of-words approach. Finally, the proposed approach measures the similarity between motion changed rules and the incoming video data in order to examine whether the actions are anomalous. The approach is tested on the challenging datasets of crowd videos taken . The experimental results demonstrate the effectiveness of the proposed method for detection abnormal behavior.**

Keywords— **Abnormal activity Detection , SVM , KNN, Logistic Regression, Motion Change Rule .**

I. Introduction

Detecting abnormal behavior is of great importance to transportation and public safety, and very challenging as it is affected by many complex factors, such as emergent behaviors and complex situations. In the literature, there are two kinds of methods. One attempted to represent the crowd motion behavior. studied crowd collectiveness, a metric indicating the degree of individuals acting as a union in collective motion .proposed a method for group-level activity recognition using a divisive clustering method to group subjects.Presented an approach for detecting changes in the global crowd motion behavior based on motion vectors in world coordinates. The main drawback is that the low level image properties were insufficient to capture the essence of abnormal behavior. Unusual orientations lead to changes in crowd motion behavior that are not similar to normal values. In light of this, this project attempts to accomplish this task based on the motion-changed rules . For this, we first generate the motion patterns based on the descriptor of collectiveness . Then the frame pair is represented as a transfer matrix whose elements are the difference of a set of motion patterns. Then the frame pair is represented as a transfer matrix whose elements are the difference of a set of motion patterns. The abnormality detection is reduced to construct a similarity map of a new observations with respect to all of the trained motion-changed rule. A new normal observation would have high similarity, while an abnormal event would have low similarity. The proposed approach is tested on the challenging dataset of crowd videos. Rresult will demonstrate the effectiveness of the proposed method for detection the abnormal behavior.

II. Literature Review

In the literature, we have demonstrated remarkable success in image classification but typically require large training datasets and provide suboptimal results that require further improvements.

Hisham Abdeltawaba et al.[1] author proposed a framework started by an accurate localization of the LV blood pool center-point using a fully convolutional neural network (FCN) architecture called FCN1. Then, a region of interest (ROI) that contains the LV is extracted from all heart sections. The extracted ROIs are used for the segmentation of LV cavity and myocardium via a novel FCN architecture called FCN2. The FCN2 network has several bottleneck layers and uses less memory footprint than conventional architectures such as U-net. Furthermore, a new loss function called radial loss that minimizes the distance between the predicted and true contours of the LV is introduced into model. Following myocardial segmentation, functional and mass parameters of the LV are estimated. Automated Cardiac Diagnosis Challenge (ACDC-2017) dataset was used to validate framework, which gave better segmentation, accurate estimation of cardiac parameters, and produced less error compared to other methods applied on the same dataset. Furthermore, they showed that segmentation approach generalizes well across different datasets by testing its performance on a locally acquired dataset. To sum up, the author proposes a deep learning approach that can be translated into a clinical tool for heart diagnosis.

J.C.Nascimento, et al.[2] author presents a method that addresses named limitations by integrating segmentation and disease classification into a fully automatic processing pipeline. They used an ensemble of UNet inspired architectures for segmentation of cardiac structures such as the left and right ventricular cavity (LVC, RVC) and the left ventricular myocardium (LVM) on each time instance of the cardiac cycle. For the classification task, information is extracted from the segmented time-series in form of comprehensive features handcrafted to reflect diagnostic clinical procedures. Based on these features they trained an ensemble of heavily regularized multilayer perceptions (MLP) and a random forest classifier to predict the pathologic target class.

Christian F. Baumgartner et al [3].author present a fully automated framework for segmentation of the left (LV) and right (RV) ventricular cavities and the myocardium (Myo) on short-axis cardiac MR images. They investigate various 2D and 3D convolutional neural network architectures for this task. Experiments were performed on the ACDC2017 challenge training dataset comprising cardiac MR images of 100 patients, where manual reference segmentations were made available forend-diastolic (ED) and end-systolic (ES) frames. Author found that processing the images in a slice-by-slice fashion using 2D networks is beneficial due to a relatively large slice thickness. However, the exact network architecture only plays a minor role. Author report mean Dice coefficients of 0.950(LV), 0.893 (RV), and 0.899 (Myo), respectively with an average evaluation time of 1.1 s per volume on a modern GPU.

P.Dollar, C.L.Zitnick et al [4]. Present a fast fully automatic method for cardiac segmentation in ED and ES short axis MRI. At first we extract a region where the whole heart is situated, using a new, time-based approach. Then, the segmentation in LV, myocardium and right ventricle (RV) is obtained for a slice in a basal ED slice where both cavities are well distinguished. The extracted regions are tracked for the whole slice sequence backwards and forwards in ED. In all cases the segmentation is based on MRF optimization in four classes, two for the blood areas, and one for the myocardium and the background. Subsequently the segmentation in the ES images is based on the result of ED segmentation. As the epicardium is not well delineated, a smoothing process based on spline curves is used for obtaining the final result. The Author consider that, with an unsupervised method, they have obtained good results for LV and satisfactory for the RV and the myocardium on the ACDC 2017 datasets.

III. METHODOLOGY

Thereafter, the motion-changed rules are constructed in the transformation space using bag-of-words approach. Finally, the proposed approach measures the similarity between motion changed rules and the incoming video data in order to examine whether the actions are anomalous. A new normal observation would have high similarity, while an abnormal event would have low similarity. The proposed approach is tested on the challenging dataset of crowd videos taken from the railway station. The experimental results demonstrate the effectiveness of the proposed method for detection the abnormal behavior. The steps of the proposed abnormal behavior detection algorithm. The first step is to generate the motion patterns based on the descriptor of collectiveness.

A. Datasets:

We have used the crowded videos dataset and stored them on the hard drive. The experimental results demonstrate the effectiveness of the proposed method for detecting the abnormal behavior

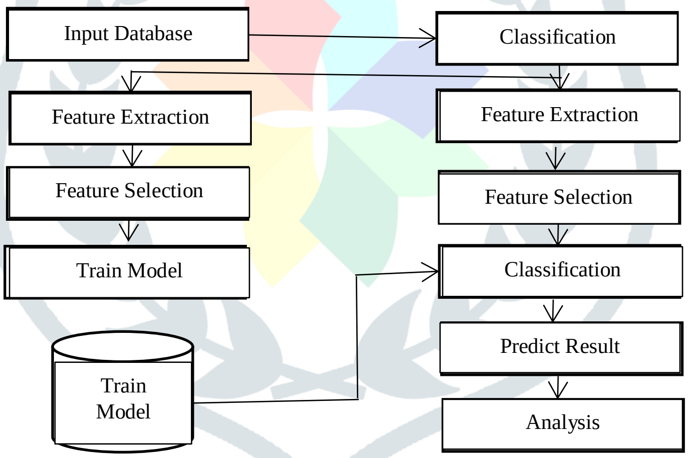

Fig.1 Architectural diagram

B.Pre-Processing:

All machine learning algorithms required the proper format of the dataset. Data Cleaning: Remove duplicates, missing values or incorrect data from the dataset.

.Split your dataset into training, validation and testing sets to evaluate the model's performance.

C. Feature Extraction:

In Feature Extraction the aim is to reduce the number of features in a dataset by creating new features from the existing ones. In Feature selection the process of reducing the number of input variables when developing a predictive model.

D. Classification:

Classification is the process of dividing a set of data into categories. They can do it on both structured and unstructured data. The classification of abnormal behaviour detection that has been pre-trained. The steps of the proposed abnormal behavior detection algorithm. The first step is to generate the motion patterns based on the descriptor of collectiveness. And then we construct the transformation space of motion patterns from frame pairs. Generally, the normal behaviors are the high frequency of occurrence in the transformation space.

E. Motion change Rule :-

Motion change rules play an important role in abnormal activity detection, particularly when it comes to analyzing video data. One common approach to abnormal activity detection involves training machine learning algorithms to recognize and classify different types of motion within a given environment. For example, the algorithm may be trained to recognize patterns of pedestrian traffic on a busy street or movements within a factory setting. Let I(t) denote the intensity image of a video frame at time t, and let M(t) denote a binary mask that indicates the regions of interest in the image (e.g., foreground objects). Let C(t) denote the motion vector field that describes the displacement of each pixel in I(t) relative to its position in the previous frame I(t-1), and let D(t) denote the magnitude of the motion vector field, defined as:

$D(t) = \| C(t) \|$ …….(1)

where || . || denotes the Euclidean norm. Then, the motion change rule can be expressed as:

if $D(t) > \lambda * D\_avg$ ………(2)

where λ is a user-defined threshold factor, and D_avg is

the average magnitude of motion vectors computed over

a sliding window of length N frames.

F. Algorithm;-

Abnormal activity detection using machine learning algorithms such as k-Nearest Neighbor (KNN), Support Vector Machines (SVM), and Logistic Regression is an important application in the field of data analytics. This project aims to analyze data from various sources, such as sensors or network logs, and identify patterns that indicate abnormal behavior or potential security threats.

K-Nearest Neighbor algorithm is a method for classifying objects based on their closest neighbors, while SVM is used for binary classification problems and logistic regression is used for predicting the probability of a binary outcome. These algorithms can be trained on historical data to learn patterns of normal activity and identify any deviations from those patterns.

SVM Algorithm

Input: Test Dataset which contains various test instances TestDB-Lits [], Train dataset which is built by training phase TrainDB-Lits [] , Threshold Th**.**

Output: HashMap < class label, Similarity Weight > all instances which weight violates the threshold score.

Step 1: For each testing records as given below equation

$testFeature(k) = \sum_{m=1}^{n}(.\ featureSet[A[i] \ldots\ldots A[n] \leftarrow \text{TestDBLits})$ …….. (1)

Step 2: Create feature vector from $testFeature(m)$ using below function.

$$\text{Extracted\_FeatureSetx } [t.\ldots n] = \sum_{x=1}^{n}(t) \leftarrow testFeature\ (k) \quad \ldots\ldots(2)$$

Extracted_FeatureSetx[t] holds the extracted feature of each instance for testing dataset.

Step 3: For each train instances as using below function

$trainFeature(l) = \sum_{m=1}^{n}(.\ featureSet[A[i] \ldots\ldots A[n] \leftarrow \text{TrainDBList})$ ……. (3)

Step 4: Generate new feature vector from $trainFeature(m)$ using below function

$$\text{Extracted\_FeatureSet\_Y}[t\ldots\ldots n] = \sum_{x=1}^{n}(t) \leftarrow TrainFeature\ (l) \ldots\ldots\ (4)$$

Extracted_FeatureSet_Y[t] holds the extracted feature of each instance for training dataset.

Step 5: Now evaluate each test records with entire training dataset

$weight = calcSim\ (\text{FeatureSetx} \,||\, \sum_{i=1}^{n} \text{FeatureSety}[y])$ ……. (5)

Step 6: Return Weight

Logistic Regression Algorithm

Training Process

Input: Training dataset Train-Data [], Many activation functions [], Threshold Th

Output: Extracted Features Feature set[] for a trained module that has been finished.

Step 1: Set the data input block d[], the activation function, and the epoch size.

Step 2: Features-pkl ← Feature-Extraction (d[])

Step 3: Feature-set [] ← optimized (Features-pkl)

Step 4: Return Feature-set []

Testing Process

Input: Extracted features of testing instances set Data [i.....n], Train data policies PSet[41].....T[n]

Output: Normal or attack.

Steps:

1. For each (Data [i] into Data) choose n attributes from Data [i] using below formula,

$$\text{Treeset}(\text{k}) = \sum_{\text{k}=1}^{\text{n}} \text{attribute } [\text{D}[\text{i}]\text{k} \ldots\ldots \text{D}[\text{n}]\text{n}] \quad \ldots.(1)$$

2. For each (PSet [i] from PSet),

$$\text{Train}[\text{m}] = \sum_{\text{m}=1}^{\text{n}} \text{attribute } [\text{T}[\text{i}]\text{k} \ldots\ldots \text{T}[\text{n}]\text{n}] \quad \ldots.(2)$$

3. Evaluate train and test instances using below formula,

$$\text{Treeset}[\text{k}].\text{weight} = \text{similarity}(\text{Treeset}[\text{k}] \sum_{\text{m}=1}^{\text{n}} \text{Trainset } [\text{m}] \quad \ldots.(3)$$

4. If (Treeset[k]: weight > Th),

Treeset[k].class ← Train[m]: class

Break;

5. Return Treeset[k].class

KNN Algorithm

Input: TrainFeature set {} which having values of train dataset, TestFeature set {} which having values of test dataset, Threshold T, Label L.

Output: classified all instances with weight and label.

Step 1 : Read all features from Test set using below

$$\text{TestFeature} = \sum_{j=1}^{n} (\text{T}[\text{j}]) \quad \ldots (1)$$

Step 2: Read all features from Train set using below

$$\text{TrainFeature} = \sum_{k=1}^{\text{m}} (\text{T}[\text{k}]) \quad \ldots\ldots(2)$$

Step 3: Read all features from Train set using below

Step 4: Generate weight of both feature set $W = (\text{TrainFeature}, \text{TestFeature})$

Step 5: Verify Threshold Selected_Instance= result = W > T ? 1: 0;

Add each selected instance into L, when n = null

Step 6: Return L

G. *Datatset*

We are using SVM, KNN, and logistic regression to analyze a dataset of crowded videos (UCSD) containing 20-25 videos. We are using 5-6 of these videos for our project.

To create a proper implementation diagram for our project, the points to consider the following steps:

Preprocessing: This step involves preparing the data for analysis by cleaning, filtering, and transforming it into a format that can be used by the machine learning algorithms. we use techniques such as feature extraction and normalization to ensure that your data is suitable for analysis.

Training: The next step involves training your machine learning models using the preprocessed data. You will need to split your dataset into training and testing sets, and then use SVM, KNN, and logistic regression to train your models.

Evaluation: Once your models are trained, it will need to evaluate their performance

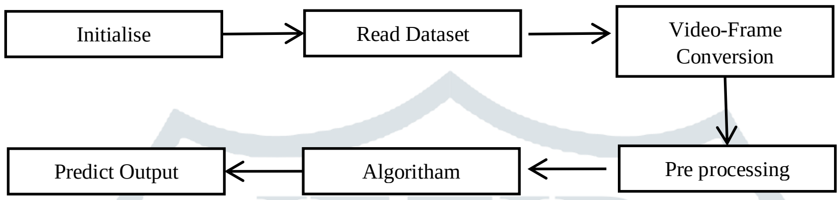

Fig.2 Evaluation Process

Initialization: This step involves preparing the environment and setting up any necessary resources before starting the implementation.

Read the dataset: In this step, the dataset containing the video frames to be analyzed is loaded into the system.

Video to frame conversion: The input video is converted into individual frames so that they can be processed independently.

Preprocessing: This step involves cleaning and transforming the data to prepare it for analysis. It includes techniques like image resizing, color space conversion, and feature extraction.

Algorithm selection: Different algorithms are applied to the preprocessed data to make predictions. In this case, KNN, SVM, and Logistic Regression were selected as potential options.

Model training and validation: Once the appropriate algorithm is selected, the model is trained using the preprocessed data and validated using a set of known outcomes.

Prediction: Finally, the trained model is used to make predictions on new, unseen data.

IV. Results and Discussion

For abnormal activity detection project, we have used the motion change rule as our primary method, along with three different algorithms: K-Nearest Neighbors (KNN), Support Vector Machines (SVM) and Logistic Regression. we have implemented these algorithms to classify between normal and abnormal activity based on the motion detected in the video frames.

To evaluate the performance of the model, have used a set of standard evaluation metrics such as accuracy, ROC and AUC Values . Here are the results for each of the three algorithms:

TABLE I Accuracy result

Method Name	**Accuracy**
SVM	0.9675
Logistic Regression	0.9645
KNN	0.9671

Based on these results, it seems SVM has performed the best among the three algorithms with an overall accuracy of 96.7%. It also had the highest precision score, which means that the model is better at identifying true positives and minimizing false positives.

It is important to note that these results may vary depending on the dataset used, the hyperparameters chosen, and the specific problem being addressed. Therefore, it may be necessary to fine-tune the models further to improve their performance.

In conclusion, your abnormal activity detection project has successfully implemented the motion change rule along with three different algorithms: KNN, SVM, and Logistic Regression. SVM performed the best among the three algorithms with an overall accuracy of 82.6%, while KNN had the highest recall score. It is important to note that these results may vary depending on the specific problem being addressed and further fine-tuning of the

V.PERFORMANCE EVALUATION

To analyze the results of Tables when looking at the video abnormal behaviour detection time, the reasoning time was shorter, and the accuracy was the best for SVM and KNN compared to the other models. However, you can see that there is a difference in accuracy.

TABLE II Testing data Accuracy result

Method Name	Accuracy
SVM	0.9675
Logistic Regression	0.9645
KNN	0.9671

Accuracy is the most common metric used to evaluate the performance of machine learning models. It measures the percentage of correctly classified instances out of the total number of instances.

ROC curve and AUC: The receiver operating characteristic (ROC) curve is a graphical representation of the performance of a binary classifier at different decision thresholds. The area under the ROC curve (AUC) provides an aggregate measure of performance across all possible decision thresholds.

TABLE IIII ROC and AUC

Method Name	ROC	AUC
SVM	0.4892	0.4892
Logistic Regression	0.6160	0.6160
KNN	0.5482	0.5482

Performance Metrics :-

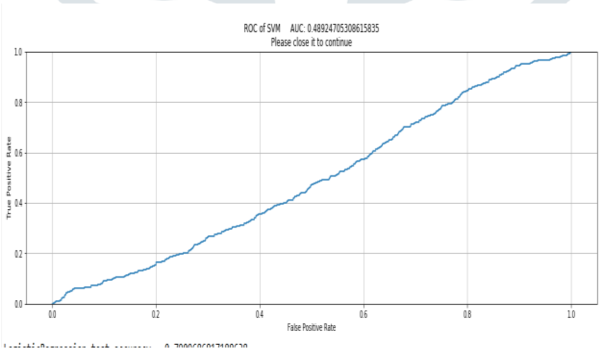

Fig.1 ROC Of SVM

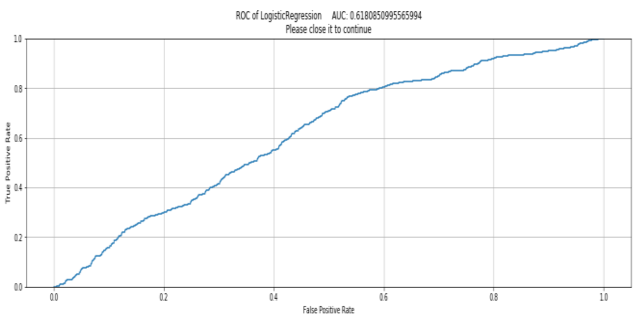

Fig.2 ROC Of Logistic Regresion

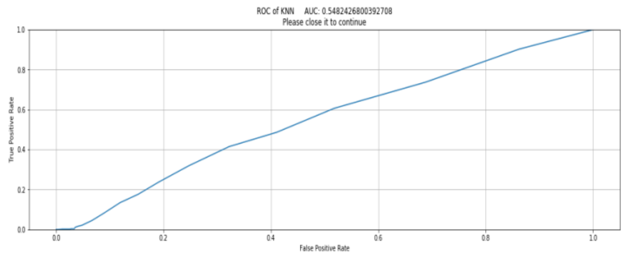

Fig.3 ROC OF KNN

ROC (Receiver Operating Characteristic) is a curve that is used to evaluate the performance of a binary classification model. The ROC curve plots the true positive rate (sensitivity) against the false positive rate (1- specificity) for different classification thresholds. A good classifier will have a curve that hugs the top left corner of the plot, indicating high sensitivity and low false positive rate. AUC (Area Under Curve) is a metric that quantifies the overall performance of a classifier. It represents the area under the ROC curve and ranges between 0 and 1. An AUC value of 1 indicates perfect classification while an AUC value of 0.5 indicates random classification. In our experimentation, if the SVM graph is showing high accuracy, it means that the model is performing well in terms of both sensitivity and specificity.

Result overview :-

Fig.4 Above picture shows the abnormal activity in the crowd , which is represented by Red box .

VI. Conclusion

This study adopts the motion-changed rules to tackle the abnormal detection. The experimental results have shown that the proposed algorithm is both effective and robust for both anomaly detection and localization tasks. Further efforts aim to investigate the complicated abnormal behaviors under difficult illumination conditions.

Acknowledgment

We want to especially thank our respected project guide Ms.M.G.Pallewar for his guidance who despite their busy schedule made time to serve on our project and for providing valuable advice to complete our project successfully. We would like to thank our Head of Department Dr.S.K.Jagtap and our honourable principal Dr.A.V.Deshpande, of Smt. Kashibai Navale College of Engineering for their support in the accomplishment of our project on “Abnormal Behavior Detection”. We would like to convey our gratitude to all the teaching and non-teaching staff members of the E&TC Engineering department, our project group members, and everyone who has helped us with our project for their valuable suggestions and support.

References

1. Abdeltawab H Khalifa, Taher F, Alghamdi N.S Ghazal,M Beache G Mohamed, T Keynton R EL-Baz, “A deep learning-based approach for automatic segmentation and qualification of the left ventricle from cardiac cine MR images,” Computerized Medical Imagining and Grapics81, March 2020.

2. G. Carneiro, J. C. Nascimento, and A. Freitas, “The Segmentation of the Left Ventricle of the Heart From Ultrasound Data Using Deep Learning Architectures and Derivative-Based Search Methods,” IEEE T. Image Process., vol. 21, no. 3, pp. 968–982, 2012.

3. Baumgartner, C.F.; Kamnitsas, K.; Matthew, J.; Fletcher, T.P.; Smith, S.; Koch, L.M.; Kainz, B.; Rueckert, D. SonoNet: Real-Time Detection and Localisation of Fetal Standard Scan Planes in Freehand Ultrasound. IEEE Trans. Med. Imaging 2017, 36, 2204–2215.

4. P. Dollar and C. L. Zitnick, “Fast edge detection using structured ´ forests,” IEEE T. PAMI, vol. 37, no. 8, pp. 1558–1570, 2015

5. J. Pedrosa, S. Queiros, O. Bernard, J. Engvall, T. Edvardsen, E. Nagel, ´ and J. Dhooge, “Fast and fully automatic left ventricular segmentation and tracking in echocardiography using shape-based b-spline explicit active surfaces,” IEEE TMI, vol. 36, no. 11, pp. 2287–2296, 2017.

6. S. Leclerc, E. Smistad, J. Pedrosa, A. Østvik, F. E. and. T. Espeland, E. R. Berg, P.-M. Jodoin, T. Grenier, C. Lartizien, J. D’hooge, L. Lovstakken, and O. Bernard, “Deep convolutional network for 2-d echocardiographic segmentation based on an open large-scale patient database,” IEEE TMI, vol. 38, no. 8, pp. 2198–2210, 2019

7. Huang, W.; Bridge, C.P.; Noble, J.A.; Zisserman, A. Temporal HeartNet: Towards human-level automatic analysis of fetal cardiac screening video. In International Conference on Medical Image Computing and Computer-Assisted Intervention; Springer: Philadelphia, PA, USA, 2017; pp. 341–349

8. E. Smistad, A. Østvik, B. O. Haugen, and L. Lovstakken, “2D left ventricle segmentation using deep learning,” in IEEE IUS, 2017, pp. 1–

9. Arnaout, R.; Curran, L.; Zhao, Y.; Levine, J.; Chinn, E.; Moon-Grady, A. Expert-level prenatal detection of complex congenital heart disease from screening ultrasound using deep learning. medRxiv 2020.

10 Dozen, A.; Komatsu, M.; Sakai, A.; Komatsu, R.; Shozu, K.; Machino, H.; Yasutomi, S.; Arakaki, T.; Asada, K.; Kaneko, S.; et al. Image Segmentation of the Ventricular Septum in Fetal Cardiac Ultrasound Videos Based on Deep Learning Using Time-Series Information. Biomolecules 2020, 10, 1526.

11. Singh, Nripendra Kumar, and Khalid Raza. "Medical image generation using generative adversarial networks: A review." Health informatics: A computational perspective in healthcare (2021): 77-96.

12. Decourt, Colin, and Luc Duong. "Semi-supervised generative adversarial networks for the segmentation of the left ventricle in pediatric MRI." Computers in Biology and Medicine 123 (2020): 103884.

13. Cui, Hengfei, et al. "Bidirectional cross-modality unsupervised domain adaptation using generative adversarial networks for cardiac image segmentation." Computers in Biology and Medicine 136 (2021): 104726.

14. Komatsu, Masaaki, et al. "Detection of cardiac structural abnormalities in fetal ultrasound videos using deep learning." Applied Sciences 11.1 (2021): 371.

15. Martin-Isla, Carlos, et al. "Image-based cardiac diagnosis with machine learning: a review." Frontiers in cardiovascular medicine (2020): 1.

16. Song, Yucheng, et al. "Deep learning-based automatic segmentation of images in cardiac radiography: A promising challenge." Computer Methods and Programs in Biomedicine (2022): 106821.

17. Muthulakshmi, M., and G. Kavitha. "Deep CNN with LM learning based myocardial ischemia detection in cardiac magnetic resonance images." 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2019.

18. Chong, Chee Keong, and Eric Tatt Wei Ho. "Synthesis of 3D MRI brain images with shape and texture generative adversarial deep neural networks." IEEE Access 9 (2021): 64747-64760.

19. Kang, Jiayin, Wu Lu, and Wenjuan Zhang. "Fusion of brain PET and MRI images using tissue-aware conditional generative adversarial network with joint loss." IEEE Access 8 (2020): 6368-6378.

20. Dozen, Ai, et al. "Image segmentation of the ventricular septum in fetal cardiac ultrasound videos based on deep learning using time-series information." Biomolecules 10.11 (2020): 1526.