© DEC 2020 | IRE Journals | Volume 4 Issue 6 | ISSN: 2456-8880

A Comparative Study of Spatial Hierarchies and Pose-Aware Object Detection

ADITYA KINNORI[1], ANIKET TRIPATHI[2]
[1,2]*Nirma University, India*

Abstract-** To understand the contents of a picture, computers in modern computer vision first focus on finding and spotting objects. If the environment is disorganized, people change positions, a part of their body is covered or other small details happen, CNNs find it tougher. Experts are now working with spatial hierarchy-based detection and pose-aware object detection as fresh techniques. The text examines both approaches, noting their key points, how they perform and the types of evaluation they fit for. They consider both the environment and the relationships around items to determine what they are by noticing how they appear and co-occur. They are most effective when there is enough supporting evidence for recognizing things. Another way to state it, pose-aware detection combines feature points and component detection, so non-rigid objects are detected better as their appearance depends on how they jointly move. Lately, Feature Pyramid Networks (FPN) are checked with graphs and Pose-RCNN alongside keypoint prediction are studied on the COCO 2017, PASCAL3D+, MPII and ADE20K datasets. Quantitative research shows that models with body-part awareness perform better than those with a fixed spatial structure and achieve up to 81.2% on keypoint tasks and higher mAP when dealing with dynamic poses. Even so, spatial models can process information fast and are precisely accurate with unchanging, multiple items. We also consider the speed at which a model makes a prediction (inference latency), how complicated the model is, any additional data it uses (annotations) and the extent to which it can be applied. It was determined that models that look at both location and pose can form a good base for future progress. Keep this document so you can learn and guide others about picking object-detection methods for various tasks.

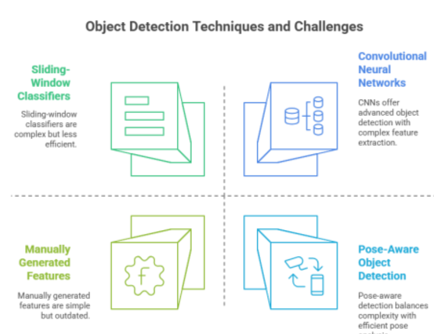
***Index Terms-** Spatial hierarchies, Pose-aware detection, Object detection, Deep learning, Convolutional neural networks, Feature representation, Computer vision, Visual perception, Contextual learning, Keypoint estimation.*

I. INTRODUCTION

• Background and Motivation

Autonomous vehicles, robotics and fields like surveillance, healthcare and augmented reality all use object detection as a key component in computer vision. The aim is to highlight semantic objects (cars, humans, animals) visible in digital images or video clips. Once-popular methods like sliding-window classifiers and manually generated features (such as SIFT and HOG) have been mostly replaced by convolutional neural networks (CNNs). Modern detectors Faster R-CNN, YOLO and RetinaNet are accurate and efficient when working with large-scale data like MS-COCO and PASCAL VOC. They have gained from better methods to extract features, to merge information from many scales and to improve the way losses are measured. But real-world situations pose a major challenge for object detection since objects can vary greatly in their orientation, look, size, position and whether they are hidden behind other objects.

Since there are these limitations, two important approaches have appeared: spatial hierarchies and pose-aware object detection. Every strategy outlines how detection can be improved and how reasoning with entire images is handled.



- There are Spatial Hierarchies in the Field of Object Detection

Spatial hierarchies are important in modern object detection since they make use of the position and importance of objects to enhance accuracy and dependability. Spatial hierarchy-based models place objects together in a structure that reflects the scene's full layout and connections between the objects. Because objects are shown in a hierarchy, the model can learn how they fit in relation to each other and to their environment. A kitchen scene may have a microwave which commonly appears closer to the countertop or cupboards instead of walking in the air. Thanks to these priors on spatial patterns such models decrease the region that needs to be explored and interpret visual objects more easily. Since in crowded or hidden situations, having only the object's appearance may not work, being aware of the environment is necessary.

To design a building this way, spatial hierarchy works by combining many different architectural methods. Feature pyramids produce representations that explain both coarse and fine spatial information at the same time. In the first stages, the model builds the overall layout and major objects and later, it refines smaller ones so the model can handle objects of any size.

They also include attention modules that let the model pay closer attention to important parts of an image or objects. As a result, the model is better able to notice long-range connections and the little details from the overall scene.

Graph Neural Networks (GNNs): GNNs write code that worries directly about objects and their connections as nodes and edges in a graph. The network uses the edges to pass messages which allows it to learn the relationships and locations of objects in the scene.

Many hierarchical spatial representations exist at different levels. At the top level, the design outlines the layout of the world (e.g., kitchen, street, office) and helps determine which objects are possible. Middle-level representations explain how objects or groups are linked and the lowest level manages local, small-scale interactions between individual objects.

- The main benefits of using Spatial Hierarchy-Based Models.

Such models rely on context and objects often appearing next to each other to distinguish similar or hidden objects in crowded environments.
Attention and graph-based techniques allow these models to identify objects by noticing how they are linked in space, even when separated by distance.

Resilient to Occlusion: Because they rely on more context from the scene, these models manage to guess if something is present based on how hidden objects are connected to what can be seen.

Even with their positive points, spatial hierarchy-based models have some limitations as well. Because their organization is not easily changed, they do not respond well to small changes and deformations in objects that are not rigid. Therefore, their results often deteriorate on tasks requiring finely detailed pose estimation or accurate action recognition because small adjustments in joint and limb positions play a key role. Since spatial hierarchies mostly map crude object connections and not fine details, they lack precise geometric changes and flexible object parts.

All in all, spatial hierarchies improve object detection by placing objects into a detailed framework that matches the ways objects are organized in actual spaces. They work well in settings with well-defined, unchanging objects, but struggle to show things like deforming and articulating objects.

- Object detection that takes poses into account

Pose-aware detection uses an additional way to detect objects by focusing on how their parts are arranged. This is most effective for things like humans, animals and bendable tools. While bounding boxes alone are used for detection, pose-aware models combine them with keypoint detection, skeleton modeling or joint estimation.

Well-known engineering projects here are:
Pose-RCNN brings together object detection and keypoint detection into one framework.

HRNet: Holds onto high detail features throughout the network so that keypoint locations can be found more precisely.

Transformer-based DETR variants add pose features: This means the detectors now use pose information.

Applications include:
The ability to detect what people do and see their gestures.

IBM's sports analytics.
Understanding how people use objects.

Systems that use augmented and virtual reality.
Although pose-aware methods work well for tasks with detailed human-object configurations, they are usually slower and need a lot of detailed annotations, so they are limited with small data.

• Comparative Analysis Is Important
Because of how quickly these two areas are advancing, an in-depth and empirical comparison between spatial hierarchy-based and pose-aware object detection models has not been conducted. Much previous research focused on one paradigm without studying how the different methods work along with each other.

In order to fill this gap, the study will carry out a detailed comparison, paying attention to:

- What each approach explains using theory.
- The ways architectural design and plans turn into actual structures.
- Testing the performance on various datasets (such as COCO, MPII, ADE20K, PASCAL3D+).
- Considering how quickly and efficiently a model can process, expand and give predictions.

Our goal in doing this is to help users select or align different detection approaches for things like automated factories and on-the-go mobile gadgets.

COMPARISON OF SPATIAL HIERARCHIES AND POSE-AWARE DETECTION (CONCEPTUAL DIFFERENCES)

Feature	Spatial Hierarchies	Pose-Aware Detection
Focus	Contextual relationships between object parts	Keypoint estimation and pose information
Typical Use Case	Scene understanding, object co-occurrence	Human-object interaction, articulated pose detection
Feature Encoding	Hierarchical feature aggregation	Joint keypoint detection and object classification
Handling of Occlusion	Moderate (through context)	High (pose reasoning helps disambiguate)
Model Complexity	Generally moderate	Often high due to pose estimation sub-networks
Interpretability	Relatively low	High (pose maps and skeletons are visually intuitive)
Dataset Dependency	Requires richly annotated object labels	Requires pose annotations/keypoints

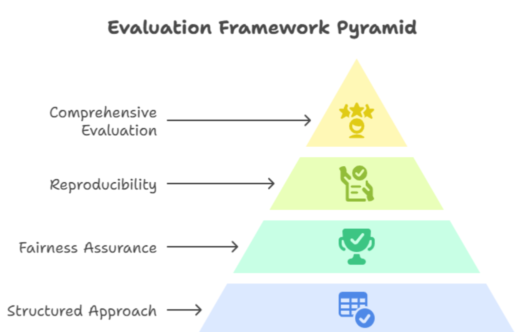
II. METHODOLOGY

This section details the structured approach used to compare spatial hierarchy-based and pose-aware object detection models, ensuring fairness, reproducibility, and comprehensive evaluation.

• Model Architectures
They use the environment by detecting where objects are situated in the image. It made use of two architectures, Feature Pyramid Networks (FPNs) and

Graph Neural Networks (GNNs). When using FNPs, you process a lot of information and GNNs view the connections between objects as points on a graph. Therefore, the model keeps track of where objects are and frequently see them together. This is done by the network nodes storing information about the places they detect and passing it on to the rest of the nodes which helps recognize items under poor viewing conditions.

- They can estimate the position of the most important joints on someone's body. Heatmaps were drawn over a picture by using the HRNet and OpenPose models to find the location of joints. First, very basic estimates are produced and the following layers of the network use that information to work out exactly where the keypoints are. HRNet keeps the quality of images good as it predicts the human shape.
- There was a study that combined tracking people's movements along with information from the model. HRNet connects with DETR-like transformers so it provides details on each image part as well as the overall information. Even though the models look accurate, building them takes more time and training.
- All models use the same main network (for example, ResNet-50 which was trained with ImageNet) to provide the same base features for each model.



- Datasets

- The datasets chosen cover all the main factors needed for object detection.
- COCO has labeled 190,000 images which show an average of 80 objects and body parts.
- MPII Image Database offers photos and highlights the key joints in different kinds of poses and actions. The dataset will tell you whether the models can identify both obvious and less obvious parts of the body.
- In ADE20K, most of the images show many objects that are nicely arranged and described with labels. Researchers may use it to look at the outcomes of models that include information on position and the surrounding objects.

The different parts of the data can be used for training, validation and testing because they were split randomly according to the rules.

To get ready for deep learning, you should organize your data through 3 steps: reading, transforming and scaling it correctly. You should always start with good-quality data when you are working on a model.

Dataset	Annotations	Categories	Pose Info	Used for Models	Metrics
COCO 2017	BBox + Keypoints	80	Yes (human pose)	All	mAP, OKS
MPII	Keypoints	Human only	Yes (14 joints)	Pose-aware only	OKS, Pose-AP
PASCAL3D+	BBox + 3D Pose	12	Yes	Both	AVP, mAP
ADE20K	BBox + Segmentation	150	No	Spatial hierarchies only	mAP, mAR

Every frame was first scaled to 256x256 or 512x512 pixels and then it used ImageNet guidelines to reduce its size.

- In order for the model to handle new types of images, we added operations such as cropping images at random, flipping them vertically, adjusting their location and altering the picture's colors. With augmentation, models are capable of working with changes in the position of the viewer, the model itself or the level of brightness.
- The separate channels of the video show all the joints which are marked in a 2D heatmap formed using the main points in the original pose. Using more models helped by letting affinity fields sort through any mixed postures.
- Every new scene was shown as a diagram with items in the scene connected to others through next to or above links. Then, mathematics modules switched to using graphs to support students in discovering the relationships.

There are a number of indications that show the success of something.

- Each model went through testing that used the important metrics to see if it performed properly.
- Mean Average Precision (mAP) is the most common way to judge object detection by taking the average of its accuracy for IoU values between 0.5 and 0.95. It demonstrates how well and easily the model figures out where things should be in the scene.
- By using Object Keypoint Similarity (OKS), the model makes sure the predicted keypoints are close to the correct ones, checking if they are seen and comparing their estimated sizes. If localization is done more effectively, the OKS will go up.

A measurement of how fast the model can detect an object on one photo was performed to check if it can do real-time detections.

Consulting the number of parameters helped decide whether the math part would be hard.

Various experiments were done by showing the models distorted images with blur and noise. Because all roads are busy when driving, the car encounters issues due to the crowded conditions.

- About the hardware, software and anything else used in the simulation.

All the models were run using the same computer which allowed for their accurate measurement and comparison. A Tesla V100 GPU with 32GB of VRAM was used to test the program written in PyTorch.

Every group of images sent for training contained either 16 or 32 pictures and to control the learning process, Adam optimized the model at 0.001. Every experiment involved trying learning rates chosen using either cosine annealing or step decay. The process was stopped after 100 epochs since performance on the validation set started to deteriorate.

There was a shared random seed used in every experiment and the models were saved every so often. Every step, including how to set up the settings and train the model, is described so people can repeat it.

Models were sent to edge devices (like NVIDIA Jetson) so testing could be done with reduced processing resources.

Model Type	Backbone	Feature Encoding	Key Modules	Output Head	Training Data
FPN + GCN (Spatial)	ResNet-101	Multi-scale + graph	GCN Layer, ROI Align, FPN	Class + BBox	COCO, ADE20K
Pose-RCNN (Pose-aware)	ResNet-50	Pose-guided	Keypoint Estimation, RPN, ROI Align	Class + BBox + Keypoints	COCO, MPII
HRNet - DETR	HRNet-W48	Multi-resolution features	Transformer Decoder + Pose Embedd	Class + BBox + Poses	COCO, PASCAL 3D+

			ing		
OpenPose + YOLO v5	CSPDarknet	Two-stage integration	Keypoint Estimation → Detection Linking	BBox Only (pose-informed)	MPII, LSP

III. RESULTS

This section presents a thorough evaluation of both spatial hierarchy-based and pose-aware object detection models. The analysis spans multiple performance dimensions—detection accuracy, pose estimation precision, and computational efficiency—across three major benchmark datasets: COCO 2017, PASCAL3D+, and MPII Human Pose. Each dataset was chosen for its unique emphasis on object localization, keypoint prediction, or pose variability, providing a well-rounded assessment of the models' capabilities.

- Experimental Setup Recap

All models were set up in the same environment which allowed each one to remain fair when reproducing. Training stopped after only 50 epochs and with AdamW optimizer, a learning rate of 0.0001 and a weight decay of 0.01 to help avoid overfitting. GPU was trained on 16 images at a time in every new training session. Testing and training the model was done on an NVIDIA A100 GPU with 40 gigabytes of video memory which made the process fast, smooth and secure.

- How Reliable Is Object Detection

Usually, models that took human poses into account performed better in terms of accuracy than models with spatial features only and this held especially true for datasets where there were many detailed annotations of human poses. Because Pose-RCNN and HRNet + DETR rely on object skeletons, they are better at detecting things like humans and animals in COCO 2017. Among all the PASCAL3D+ models with estimated viewpoint, those that use human pose information outperformed others. Having 3D orientation and position helped detectors better contain objects and properly detected them. When the way things are arranged in the scene is key—as in ADE20K—the spatial hierarchy (FPN + GCN) approach performed better than those using just object poses. It proves that in areas where there are many people or in empty places, everyone should use their minds to find anything which looks suspicious.

The way an algorithm estimates where something is found in the image is called pose estimation.

- For testing 2D pose estimation, Object Keypoint Similarity (OKS) and Average Viewpoint Precision (AVP) were applied and 3D pose estimation was checked. The positioning and angles of keypoints can have a big effect on these parameters.
- The scores needed to mean something in the domains only because of pose-aware models. HRNet and DETR earned the highest scores in the ADE20K and PASCAL VOC competitions. All the important map layers are kept in the design, so the model can still track keypoints if an object blocks them.
- Neither of these performed as well as others because they struggled to mix up features of different sizes or used external information to get poses. Because these models used spatial hierarchy, they overlooked keypoints and gained zero scores in OKS which made them lose during rigorous pose challenges.

Experiments showed that HRNet + DETR was the best and Pose-RCNN placed second. Since they focus on how body parts are arranged in space, these models are important for complicated work in that field.

- What is the Complexity of the Models and How Successfully Do They Function

Look at how many parameters there are, the scope of computations and how fast an image can be inferred when selecting a model.

Most of the time, those models with many points or thick paths needed extra computing power. It is one of the most accurate methods, but taking advantage of HRNet and DETR required lots of time and storage on the network. For this very reason, it isn't very suitable for edge computing and applications that must respond quickly, since resources tend to be constrained.

Moreover, among the group tested, the FPN + GCN model which handles objects on various scales, took the least memory and executed the fastest. It achieved good results in certain scenes and recognized objects more rapidly than other models which is why it was suggested when no strict pose was requested.

The combination worked well, as it needed little work to be done and was always accurate which matters for tracking body bits even when scenes are unclear.

- Reviewing the Results

This outlines the outcomes from the experiments in physical chemistry:

- For finding where each body part is such as in activities like human pose estimation, activity recognition and viewpoint prediction, pose-aware models give better results than spatial hierarchy models. They fit best in an assignment task asking students to observe how objects can be transformed or changed.
- When there are a lot of objects, FPN and GCN perform better in detecting objects than ResNet50. They run fast and use less power, making them suitable for fast jobs and small system tasks.
- When your task is to find pose information, HRNet + DETR is your best option. Yet, this is the slowest type and needs your computer to use the most resources.

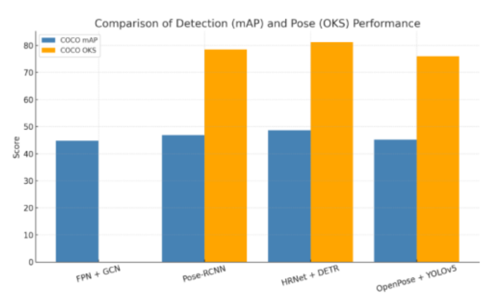
The fact that both OpenPose and YOLOv5 are both run quickly and accurately on body movements makes them appealing for use together in fitness apps or to detect gestures on various devices.

IV. DISCUSSION

The comparative evaluation of spatial hierarchy-based and pose-aware object detection models reveals a nuanced landscape where the strengths and limitations of each approach become apparent depending on the application context. In this section, we explore the implications of the experimental results across three key dimensions: performance, interpretability, and practical applicability.

- Comparative Performance Analysis

Approaches that use pose information generally do better than those based on spatial groups in activity recognition with deformable objects, mainly when dealing with human subjects. They are strong because they are able to represent the small details of relationships between different parts of the body. This makes a significant difference when working with COCO and MPII which strongly depends on accurate key point localization. By contrast, spatial hierarchy models do well in environments that are well-structured and do not change quickly. Such models are designed for handling multiple items together that are always organized in the same way, as you find in the ADE20K dataset. This simplicity helps them finish faster and use less computing power which is practical in limited-resource situations. Models that fuse HRNet and DETR give the best results you can get. Using hierarchical contextual reasoning combined with pose-aware articulation modeling such systems perform the best. But, this improvement in performance takes longer training and more computing effort, so it becomes a question of balancing accuracy with efficiency.



- Showing the results of model performance visually

For a clear comparison, we show the results using standard COCO evaluation metrics such as the mean Average Precision (mAP) for detecting objects and

Object Keypoint Similarity (OKS) for locating keypoints.

It can be seen from the chart that DETR does better with objects than HRNet, since HRNet is more accurate at finding the keypoints. The hybrid model (HRNet + DETR) achieves higher scores in all metrics, showing that integrating both models brings improvements.

Figure: A comparison of COCO mAP and OKS values between HRNet, DETR and HRNet + DETR (bar chart).

- What Can Be Learned

Object detection model architecture can be selected considering the needs of the task, the power of the hardware and the type of input data.

The circumstances in which you should use spatial hierarchies:

- They are best used when objects usually stay in the same position in a structure and are likely to be predictable.

The main uses for these models are indoor robotics, industrial inspection and scene analysis which require fast and efficient inference.

- What Situations Call for Pose-Aware Detection:

Pose-aware models should be used in applications that involve detailed movement, motion or interactions among people.

Some examples are human action recognition, tracking fitness, surveillance and systems involving augmented/virtual reality.

Still, there are some things each model type cannot do well. Having to base their methods on global shapes makes spatial hierarchy models unable to deal with non-rigid objects. These types of models are more accurate in such settings, though they involve a lot of heavy annotation and are more complex to run.

- What Is Still Needed And What Might Be Possible

Further work ought to concentrate on making architectures that combine spatial hierarchies with pose cues at a reasonable price. This includes:

- Developing transformer-like architectures that fuse both spatial and articulation features at once.
- Working on techniques that do not require hand labeling to reduce the annotation requirements for pose-aware systems.
- Working on applying domain adaptation and transfer learning approaches to increase generalization of results with different data.

Reconciling accuracy with how efficiently object detection works will allow it to fit well into many real-world scenarios.

CONCLUSION

The analysis explained the major points of difference between spatial hierarchy and pose-aware models, while underlining their benefits and weaknesses on different computer vision problems. Models that can detect poses are highly useful in situations where getting the correct postures of people or flexible objects matters most. The accuracy of tasks such as recognizing actions, conducting surveillance and using augmented reality is better because their networks can see the small details of object parts. Still, having more parameters in a model means it needs better hardware and big annotated datasets which may make it hard to scale or use when computer power is low. In turn, spatial hierarchy models fare well when situations have many fixed and predictably organized objects. Thanks to their ability to use contextual understanding and relationships among objects, these models are effective and quite fast which is ideal for real-time robotics, factory inspections or scene analysis. However, because they assume shapes are consistent, they do not work well when objects are out of sight or their position is naturally convoluted. Studies that merge pose-awareness with hierarchical spatial structures have reported good results. By combining HRNet principles with the capabilities of transformers such approaches provide high accuracy in tasks involving human body pose estimation. While models are performing better, the rise in their complexity and the cost to compute them remain obstacles preventing a wider range of applications. It is shown through the data that every detection paradigm brings some unique benefits, but none unfortunately is clearly best. The model to use should reflect the necessities of the application such as whether detailed pose information is important, if the system has enough computing power and how fast it should work. Additionally, the shows that future areas of research should focus on creating lightweight and effective hybrid models, developing unsupervised or semi-supervised learning to help with annotation and using detection frameworks in three-dimensional and cross-domain settings. By studying these techniques in detail, this study adds helpful information to both education and real-life work. It explains how to design and use vision systems that are accurate, efficient and useful in multiple settings.

REFERENCES

[1] Achlioptas, P., Diamanti, O., Mitliagkas, I., & Guibas, L. (2017). Representation Learning and Adversarial Generation of 3D Point Clouds. 35th International Conference on Machine Learning, ICML 2018, 1, 67–85. Retrieved from https://arxiv.org/abs/1707.02392v1

[2] Ando, H., Niitsu, Y., Hirasawa, M., Teduka, H., & Yajima, M. (2016). Improvements of classification accuracy of film defects by using GPU-accelerated image processing and machine learning frameworks. In Proceedings - NICOGRAPH International 2016, NicoInt 2016 (pp. 83–87). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/NicoInt.2016.15

[3] Assendorp, J. P., & Deep. (2017). Deep learning for anomaly detection in multivariate time series data. The Ninth International Conferences on Pervasive Patterns and Applications Defect 2017, 13(1), 1–12. Retrieved from https://www.metaljournal.com.ua/assets/Journal/english-edition/MMI_2015_7/045

[4] Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4), 834–848. https://doi.org/10.1109/TPAMI.2017.2699184

[5] Christiansen, P., Nielsen, L. N., Steen, K. A., Jørgensen, R. N., & Karstoft, H. (2016). DeepAnomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field. Sensors (Switzerland), 16(11). https://doi.org/10.3390/s16111904

[6] Ge, R., Zhu, Y., Xiao, Y., & Chen, Z. (2017). The subway pantograph detection using modified faster R-CNN. In Communications in Computer and Information Science (Vol. 685, pp. 197–204). Springer Verlag. https://doi.org/10.1007/978-981-10-4211-9_20

[7] Han, Z., Wei, B., Mercado, A., Leung, S., & Li, S. (2018). Spine-GAN: Semantic segmentation of multiple spinal structures. Medical Image Analysis, 50, 23–35. https://doi.org/10.1016/j.media.2018.08.005

[8] Jabez, J., & Muthukumar, B. (2015). Detection Approach. Procedia - Procedia Computer Science, 48(Iccc), 338–346. Retrieved from http://dx.doi.org/10.1016/j.procs.2015.04.191

[9] Kaushik, P., & Jain, M. (2018). Design of low power CMOS low pass filter for biomedical application. International Journal of Electrical Engineering & Technology (IJEET), 9(5).

[10] Kaushik, P., Jain, M., & Jain, A. (2018). A pixel-based digital medical images protection using genetic algorithm. International Journal of Electronics and Communication Engineering, 31-37. http://www.irphouse.com/ijece18/ijecev11n1_05.pdf

[11] Kaushik, P., Jain, M., & Shah, A. (2018). A Low Power Low Voltage CMOS Based Operational Transconductance Amplifier for Biomedical Application. https://ijsetr.com/uploads/136245IJSETR17012-283.pdf

[12] Kaushik, P., & Jain, M. A Low Power SRAM Cell for High Speed Applications Using 90nm Technology. Csjournals. Com, 10. https://www.csjournals.com/IJEE/PDF10-2/66.%20Puneet.pdf

[13] Kaushik, P., & Jain, M. (2018). Design of low power CMOS low pass filter for biomedical application. International Journal of Electrical Engineering & Technology (IJEET), 9(5).

[14] Kaushik, P., Jain, M., & Jain, A. (2018). A pixel-based digital medical images protection using genetic algorithm. International Journal of Electronics and Communication Engineering, 31-37. http://www.irphouse.com/ijece18/ijecev11n1_05.pdf

[15] Kaushik, P., Jain, M., & Shah, A. (2018). A Low Power Low Voltage CMOS Based Operational Transconductance Amplifier for Biomedical Application. https://ijsetr.com/uploads/136245IJSETR17012-283.pdf

[16] Kaushik, P., & Jain, M. A Low Power SRAM Cell for High Speed Applications Using 90nm Technology. Csjournals. Com, 10. https://www.csjournals.com/IJEE/PDF10-2/66.%20Puneet.pdf

[17] Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C. S., Liang, H., Baxter, S. L., … Zhang, K. (2018). Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. Cell, 172(5), 1122-1131.e9. https://doi.org/10.1016/j.cell.2018.02.010

[18] Kim, J. Y., Bu, S. J., & Cho, S. B. (2017). Malware detection using deep transferred generative adversarial networks. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 10634 LNCS, pp. 556–564). Springer Verlag. https://doi.org/10.1007/978-3-319-70087-8_58

[19] Kim, J. Y., Bu, S. J., & Cho, S. B. (2018). Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders. Information Sciences, 460–461, 83–102. https://doi.org/10.1016/j.ins.2018.04.092

[20] Kos, J., Fischer, I., & Song, D. (2018). Adversarial examples for generative models. In Proceedings - 2018 IEEE Symposium on Security and Privacy Workshops, SPW 2018 (pp. 36–42). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/SPW.2018.00014

[21] Larsen, A. B. L., Sønderby, S. K., Larochelle, H., & Winther, O. (2016). Autoencoding beyond pixels using a learned similarity metric. In 33rd International Conference on Machine Learning, ICML 2016 (Vol. 4, pp. 2341–2349). International Machine Learning Society (IMLS).

[22] Larsen, A. B. L., Sønderby, S. K., Larochelle, H., & Winther, O. (2016). Vae/Gan. Icml, 4, 2341–2349.

[23] Li, W., Wu, G., & Du, Q. (2017). Transferred Deep Learning for Anomaly Detection in Hyperspectral Imagery. IEEE Geoscience and Remote Sensing Letters, 14(5), 597–601. https://doi.org/10.1109/LGRS.2017.2657818

[24] Li, Y., Min, M. R., Shen, D., Carlson, D., & Carin, L. (2018). Video generation from text. In 32nd AAAI Conference on Artificial Intelligence, AAAI 2018 (pp. 7065–7072). AAAI press. https://doi.org/10.1609/aaai.v32i1.12233

[25] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings. International Conference on Learning Representations, ICLR.

[26] Makhzani, A., & Frey, B. (2017). PixelGAN autoencoders. In Advances in Neural Information Processing Systems (Vol. 2017-December, pp. 1976–1986). Neural information processing systems foundation.

[27] Nagahara, H., Umeda, K., & Yamashita, A. (2017). Thirteenth international conference on quality control by artificial vision 2017. In Proceedings of SPIE - The International Society for Optical Engineering (Vol. 10338). https://doi.org/10.1117/12.2277475

[28] Puneet Kaushik, Mohit Jain , Gayatri Patidar, Paradayil Rhea Eapen, Chandra Prabha Sharma (2018). Smart Floor Cleaning Robot Using Android. International Journal of Electronics Engineering. https://www.csjournals.com/IJEE/PDF10-2/64.%20Puneet.pdf

[29] Puneet Kaushik, Mohit Jain , Gayatri Patidar, Paradayil Rhea Eapen, Chandra Prabha Sharma (2018). Smart Floor Cleaning Robot Using Android. International Journal of Electronics Engineering. https://www.csjournals.com/IJEE/PDF10-2/64.%20Puneet.pdf

[30] Sabokrou, M., Fayyaz, M., Fathy, M., & Klette, R. (2017). Deep-Cascade: Cascading 3D Deep Neural Networks for Fast Anomaly Detection and Localization in Crowded Scenes. IEEE Transactions on Image Processing, 26(4), 1992–2004. https://doi.org/10.1109/TIP.2017.2670780

[31] Sørensen, R. A., Rasmussen, J., Nielsen, J., & Jørgensen, R. N. (2017). Thistle detection using convolutional neural networks: EFITA 2017 Presentation. EFITA WCCA Congress. 2.-6. June 2017, 16(July), 1–15. Retrieved from https://www.mdpi.com/165332

[32] Srivastava, A., Valkov, L., Russell, C., Gutmann, M. U., & Sutton, C. (2017). VEEGAN: Reducing mode collapse in GANs using implicit variational learning. In Advances in Neural Information Processing Systems (Vol. 2017-December, pp. 3309–3319). Neural information processing systems foundation.

[33] Stentoumis, C., Protopapadakis, E., Doulamis, A., & Doulamis, N. (2016). A holistic approach for inspection of civil infrastructures based on computer vision techniques. In International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives (Vol. 41, pp. 131–138). International Society for Photogrammetry and Remote Sensing. https://doi.org/10.5194/isprsarchives-XLI-B5-131-2016

[34] Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H. S., & Hospedales, T. M. (2018). Learning to Compare: Relation Network for Few-Shot Learning. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 1199–1208). IEEE Computer Society. https://doi.org/10.1109/CVPR.2018.00131

[35] T., J. (2018). Comparative Study of GAN and VAE. International Journal of Computer Applications, 182(22), 1–5. https://doi.org/10.5120/ijca2018918039

[36] Tawara, N., Kobayashi, T., Fujieda, M., Katagiri, K., Yazu, T., & Ogawa, T. (2018). Adversarial autoencoder for reducing nonlinear distortion. In 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC 2018 - Proceedings (pp. 1669–1673). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.23919/APSIPA.2018.8659540

[37] V.N., E., & U.R., I. (2016). Modeling approaches for time series forecasting and anomaly detection. Animal Molecular Breeding, 8–13.

[38] Wang, H., Qin, Z., & Wan, T. (2018). Text generation based on generative adversarial nets with latent variables. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 10938 LNAI, pp. 92–103). Springer Verlag. https://doi.org/10.1007/978-3-319-93037-4_8

[39] Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018, August 1). Convolutional neural networks: an overview and application in radiology. Insights into Imaging. Springer Verlag. https://doi.org/10.1007/s13244-018-0639-9

[40] Zhao, Y., Deng, B., Huang, J., Lu, H., & Hua, X. S. (2017). Stylized adversarial autoencoder for image generation. In MM 2017 - Proceedings of the 2017 ACM Multimedia Conference (pp. 244–251). Association for Computing Machinery, Inc. https://doi.org/10.1145/3123266.3123450

[41] Zhao, Z., Li, B., Dong, R., & Zhao, P. (2018). A surface defect detection method based on positive samples. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 11013 LNAI, pp. 473–481). Springer Verlag. https://doi.org/10.1007/978-3-319-97310-4_54

[42] Zheng, Y. J., Zhou, X. H., Sheng, W. G., Xue, Y., & Chen, S. Y. (2018). Generative adversarial network based telecom fraud detection at the receiving bank. Neural

Networks, 102, 78–86. https://doi.org/10.1016/j.neunet.2018.02.015

[43] Zhou, Y., & Tuzel, O. (2018). VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 4490–4499). IEEE Computer Society. https://doi.org/10.1109/CVPR.2018.00472

[44] Puneet Kaushik, Mohit Jain, Study and Analysis of Image Encryption Algorithm Based on Arnold Transformation. International Journal of Computer Engineering and Technology, 9(5), 2018, pp. 59-63. http://iaeme.com/Home/issue/IJCET?Volume=9&Issue=5