iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VI **Month of publication:** June 2023

DOI: https://doi.org/10.22214/ijraset.2023.54171

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

Virtual Sketch Board with Computer Vision

Raushan Kumar[1], Advin Manhar[2]
Amity School of Engineering & Technology, Amity University

Abstract: ***As a result, there was a need for a dust-free classroom for children. Thecolor marker then used is detected and mask is produced. It's been really tough to educate scholars on an online platform and make the assignment intriguing during the COVID- 19 epidemic. We're then now, using the ways of computer vision in open cv to make this design. The required language for this design is python due to its further total libraries and easy to make use of the syntax and but understanding the basics as well as it can be enforced in any open cv supported languages The colour shadowing and discovery processes are used to achieve the thing of this design. This program uses hand gestures to give druggies with an intuitive system of Human Computer Interaction(HCI). One multicolored object at the tip of the cutlet is substantially used as the marker. HCI's major thing is to ameliorate mortal- computer commerce. Virtual Sketch is in where we can draw by just landing the stir of a multicolored marker with a camera. corrosion makes the contaminations present in the mask to get reduced and Dilation further regains the eroded main mask. The coming way of morphological operations on the mask produced those are Erosion andDilation.Virtual sketching is a fashion that allows druggies to produce digital delineations or sketches that pretend the appearance of traditional pencil or essay sketches. This can be achieved using OpenCV, an open- source computer vision library, andPython.To produce a virtual sketch, the first step is to read in an image using OpenCV's imread() function. The image is also converted to grayscale using cvtColor() function. This grayscale image will be used as the base for creating thesketch.The coming step is to apply a Gaussian blur to the grayscale image using GaussianBlur() function to smooth out any noise or irregularities in the image. This is followed by using the Canny edge discovery algorithm with the Canny() function to descry the edges of theimage.The edges detected by the Canny algorithm are also thresholded using threshold() function to produce a double image where the edges are shown in white and the background isblack.Finally, to produce the sketch effect, the double image is reversed usingbitwise_not() function, and also the performing image is blended with the original grayscale image usingbitwise_and() function. This will give the appearance of a sketch, where the edges are emphasized and the background islighter.By combining these way, a virtual sketch can be created using OpenCV and Python.***

I. INTRODUCTION

The interaction between humans and machines has mostly been through direct contact methods such as a mouse, keyboard, touch screen, and remote control. However, humans communicate primarily through non-contact methods such as sound and physical motions, and researchers have been attempting to develop ways for computers to identify human intentions and information using non-contact methods such as voice, facial expressions, and gestures. Gestures are a vital aspect of human language and communication, and they are considered the simplest way for humans and computers to communicate. Gesture recognition applications such as sign language recognition and robotics fall under the category of gesture recognition. There are two methods for gesture recognition for HCI applications: wearable or direct physical approaches and computer vision-based approaches. Wearable or direct physical approaches use sensors to capture hand motion and location, whereas computer vision-based approaches use cameras to provide contactless communication between humans and machines. However, computer vision-based approaches have certain limitations due to differences in hand sizes, hand position, hand orientation, lighting conditions, and other factors. The project uses OpenCV and Python to create a sketching on air feature, where the user can draw on the screen by moving their hand in the air in front of the webcam. OpenCV is an open-source computer vision and machine learning software that has more than 2400 algorithms, including a comprehensive set of classic and state-of-the-art computer vision and machine learning algorithms. Python is a high-level general-purpose programming language that uses an object-oriented approach to write clear, logical code for small and large-scale projects. The project uses morphological operations, a set of operations that process images based on shapes, to perform the necessary image processing operations. The tracking feature in the project helps the user draw on the screen by moving their hand in the air in front of the webcam. This real-time webcam data generated by tracking the movement of the hand helps the user to draw simple things that are both interesting and challenging. In summary, the virtual sketch application using OpenCV and Python allows for real-time drawing or sketching on a canvas using hand gestures. This gesture recognition application provides a contactless communication method between humans and machines, making it more intuitive and natural.

©IJRASET: All Rights are Reserved

The project uses morphological operations to perform image processing operations, and the tracking feature allows the user to draw on the screen by moving their hand in front of the webcam. The virtual sketch application using OpenCV and Python is an innovative solution that enables users to create and express themselves in a new way.

II. OBJECTIVE AND PURPOSE

The process of making a virtual sketch using OpenCV involves capturing an image or video feed, converting it to a grayscale format, and then applying various image processing techniques to create a digital sketch. OpenCV is a popular open-source library for computer vision and image processing, providing developers with a wide range of tools and functions for creating complex digital sketches.One of the primary objectives of making a virtual sketch using OpenCV is to create a digital representation of a real-life object or scene. This can be useful for a variety of applications, from creating digital art to designing products and visual effects for film and video. With virtual sketching, artists and designers can experiment with different styles, techniques, and materials without the constraints of physical mediums.Virtual sketching also offers a level of precision and control that traditional art and design methods may not provide. With OpenCV, developers can fine-tune their sketches by adjusting parameters such as line thickness, shading, and texture. This level of control can be especially useful for product design, allowing designers to create realistic 3D models and prototypes with a high level of accuracy.Another important objective of making a virtual sketch using OpenCV is to create a digital asset that can be easily shared and manipulated. Once a sketch has been created, it can be saved as a digital file and shared with others, allowing for collaboration and feedback. Additionally, virtual sketches can be easily modified and edited, making it easy to iterate on designs and concepts.

The purpose of making a virtual sketch using OpenCV can vary depending on the specific application. Some common purposes include:

1) *Artistic Expression:* Virtual sketching can be used as a form of artistic expression, allowing artists to experiment with different styles and techniques. With OpenCV, artists can create digital artwork, illustrations, and comics that are visually stunning and highly detailed.
2) *Design:* Virtual sketching can be a powerful tool for designers, allowing them to create 3D models and prototypes that are highly accurate and realistic. With virtual sketching, designers can experiment with different materials, textures, and lighting conditions to create products that meet the needs of their customers.
3) *Animation:* Virtual sketching can be used in the animation industry to create storyboards and concept art for movies, TV shows, and video games. With OpenCV, animators can create detailed sketches that accurately depict the movements and actions of characters, making it easier to bring them to life on screen.
4) *Visual Effects:* Virtual sketching can be used in the creation of visual effects for film or video. With OpenCV, developers can add texture and detail to scenes, creating a more realistic and immersive viewing experience for audiences.
5) *Education:* Virtual sketching can be used as a tool for teaching and learning, allowing students to experiment with digital drawing and painting techniques. With OpenCV, students can create sketches that accurately depict real-life objects and scenes, helping them to develop their skills and knowledge in a hands-on way.

In conclusion, the objective and purpose of making a virtual sketch using OpenCV is to create a digital representation of a real-life object or scene that can be used for a variety of applications, from artistic expression to design and education. With OpenCV, developers have access to a wide range of tools and functions that allow them to create complex and highly detailed sketches that are visually stunning and accurate. Whether you're an artist, designer, or educator, virtual sketching using OpenCV offers a powerful tool for creating digital assets that are both beautiful and functional.

III. PROBLEM STATEMENT

Virtual sketching with OpenCV using Python is a technology that enables users to create digital sketches that mimic traditional paper-based sketches while also providing the additional functionality of being able to manipulate the sketches in real-time. The technology combines the power of computer vision and machine learning algorithms to recognize and interpret user inputs and convert them into digital sketches that can be easily edited and manipulated. One of the main problems that virtual sketching with OpenCV using Python seeks to address is the difficulty that users often face when trying to create digital sketches using traditional drawing tools. Drawing on paper allows for a certain level of tactile feedback and physical control that is difficult to replicate using a mouse or other input device. This can make it challenging for users to create sketches that accurately capture their vision or express their ideas effectively. Virtual sketching with OpenCV using Python seeks to overcome this challenge by providing a more intuitive and user-friendly interface that allows users to draw and manipulate sketches in real-time.

By leveraging the power of computer vision and machine learning algorithms, the system is able to recognize and interpret different types of sketch strokes and convert them into digital sketches that can be easily edited and manipulated. Another key advantage of virtual sketching with OpenCV using Python is that it allows users to create sketches that are more accurate and precise than traditional paper-based sketches. By using advanced algorithms to recognize and classify different types of sketch strokes, the system is able to provide users with more accurate and precise sketches that better reflect their ideas and vision. This can be particularly useful in fields such as architecture, engineering, and product design, where accurate and precise sketches are essential for communicating ideas and concepts effectively. Virtual sketching with OpenCV using Python also has the advantage of being able to convert sketches into 3D models or other digital formats that can be used for further analysis or design purposes. This can be particularly useful in fields such as product design, where the ability to create 3D models of sketches can help designers better understand how different design elements. One of the key challenges associated with virtual sketching with OpenCV using Python is the need to develop algorithms that are able to accurately recognize and classify different types of sketch strokes. This requires a deep understanding of the way that people draw and the different types of sketch strokes that are typically used. In addition, the system must be able to handle a wide variety of different drawing styles and inputs in order to be effective across a range of different users and use cases. Another challenge associated with virtual sketching with OpenCV using Python is the need to develop a user-friendly interface that is easy to use and intuitive. This requires a deep understanding of the needs and preferences of different types of users, as well as an understanding of the way that people typically interact with digital interfaces. It also requires careful attention to detail in order to ensure that the system is able to provide users with the tools and features that they need to create effective sketches. Despite these challenges, virtual sketching with OpenCV using Python has the potential to revolutionize the way that people create and interact with digital sketches. By combining the power of computer vision and machine learning algorithms with a user-friendly interface, the technology is able to provide users with a powerful and intuitive tool for creating and manipulating sketches in real-time. Whether used for product design, architecture, or other creative endeavors, virtual sketching with OpenCV using Python has the potential to help people better communicate their ideas and express their creativity.

IV. WORKING PRINCIPAL

Below shown modules used in my project.

1) Hand landmark detection involves identifying the location of joints and fingertips on an image of a hand, resulting in the production of 22 keypoints, 21 of which are located on the hand or palm, while the 22nd point represents the background.

2) The Drawing Tools module allows users to select different drawing tools, including curve, straight line, circle, rectangle, and eraser, by creating a set of functions. Additionally, an audio module has been added that produces audio feedback corresponding to the selected tool. However, it may have a negative impact on systems with lower configurations, but it is anticipated to function well on systems with higher configurations.

A. Algorithm Diagram



Fig-1 Algorithm Diagram

B. UML Diagram

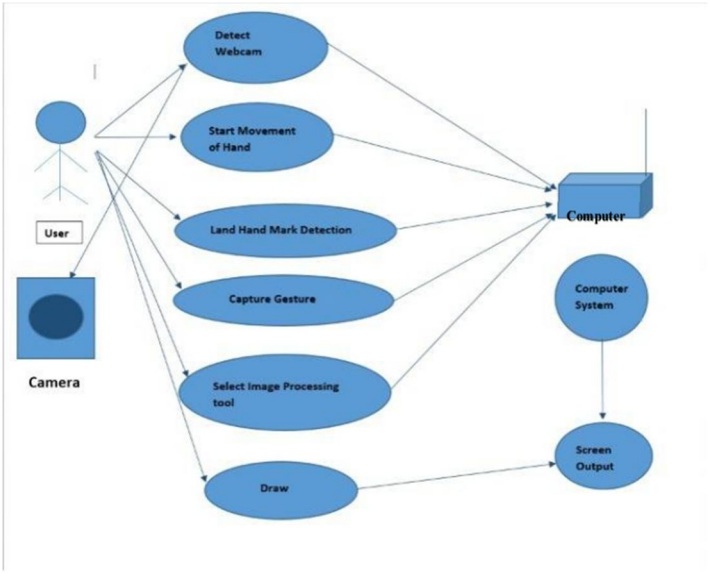


Fig-2 UML Diagram

V. PROPOSED WORK

A. Technologies Used

1) Python

a) Python is a high-level, interpreted programming language that is widely used for various purposes such as web development, data analysis, artificial intelligence, machine learning, scientific computing, and more.

b) Python has a simple syntax that is easy to learn and read, making it a great language for beginners. It is also known for its strong support for object-oriented programming and functional programming paradigms, as well as its extensive standard library that provides a wide range of tools and modules for various tasks.

c) Python code can be run on multiple platforms including Windows, macOS, Linux, and other operating systems. It also supports multiple programming paradigms, including procedural, object-oriented, and functional programming, making it a versatile language. Some popular Python frameworks and libraries include Django, Flask, NumPy, Pandas, TensorFlow, and more.

2) PyCharm

a) It is one of the most popular IDEs for Python developers and is available in two editions: Community and Professional.

b) PyCharm provides a wide range of features and tools to enhance productivity and streamline development workflows. It includes a code editor with features such as syntax highlighting, code completion, and refactoring tools, as well as debugging and testing tools, version control integration.

c) The Community edition of PyCharm is available for free and includes basic features such as code completion and syntax highlighting, as well as support for version control systems like Git. The Professional edition includes advanced features such as remote development, web development support, database tools, and more.

d) PyCharm also supports plugins that extend its functionality, allowing developers to customize the IDE according to their specific needs. Overall, PyCharm is a powerful IDE that can greatly improve the productivity of Python developers.

3) Numpy

a) It stands for "Numerical Python" and provides powerful tools for working with arrays and matrices of numerical data.

b) NumPy provides a multidimensional array object called "ndarray", which is much faster and more memory-efficient than traditional Python lists. It also provides a wide range of mathematical functions for working with arrays, such as trigonometric functions, statistical functions, linear algebra operations, and more.

c) NumPy is widely used in many scientific computing applications, including data analysis, machine learning, image processing, and more. It also integrates well with other Python libraries such as Pandas and Matplotlib.

4) CV2

a) cv2 is a Python library for computer vision and image processing. It stands for "OpenCV", which is short for "Open-Source Computer Vision Library".

b) cv2 provides a wide range of functions and tools for working with images and videos, such as loading and displaying images, applying image filters and transformations, detecting and recognizing objects in images, and more. It also provides tools for working with video streams, such as reading video files, capturing live video from cameras, and processing video frames in real-time.

c) cv2 is widely used in many computer vision applications, including object detection and recognition, face recognition, image segmentation, and more. It also integrates well with other Python libraries such as NumPy and Matplotlib.

5) Mediapipe

a) MediaPipe is an open-source cross-platform framework for building multimodal machine learning pipelines, developed by Google. The framework provides a set of building blocks, called "graphs," that developers can use to build complex multimedia applications. These graphs consist of pre-built and customizable components that can be easily connected to build pipelines for processing multimedia data such as images, video, and audio.

b) MediaPipe provides pre-built graphs for various tasks such as face detection, hand tracking, pose estimation, and more, all built using machine learning algorithms that are optimized for real-time performance Developers can also create their own custom graphs by combining the pre-built components or by building new ones from scratch.

c) MediaPipe is designed to be highly customizable, making it possible to tailor the pipeline to the specific requirements of the application. The framework provides tools for visualizing and analyzing the results of the pipeline, as well as tools for exporting the pipeline to various platforms such as TensorFlow Lite, Android, and iOS.

d) MediaPipe also includes a Python API, which makes it easy to use the framework in Python scripts and applications. The API provides access to all the functionality of the framework, allowing developers to build, test, and deploy machine learning pipelines entirely in Python.

e) Overall, MediaPipe is a powerful and flexible framework for building multimodal machine learning pipelines, making it easier for developers to build complex multimedia applications that process data from a variety of sources.

6) OpenCv

a) OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning library that is written in C++ and can be used with Python and other programming languages. It provides a wide range of functions and tools for working with images and videos, including image and video capture, image filtering, feature detection, object recognition, and more.

b) OpenCV includes a large set of algorithms and functions for various computer vision tasks such as image and video processing, object detection and tracking, machine learning, and more. It also includes a range of utilities for processing images and videos, such as reading and writing image and video files, displaying images, and more.

c) OpenCV is widely used in many computer vision applications, including robotics, automotive, healthcare, security, and more. It also integrates well with other libraries such as NumPy and Matplotlib.

VI. SYSTEM REQUIREMENTS

1) Operating System: Windows

2) IDE: Jupyter Notebook, Anaconda, PyCharm, NumPy etc.

3) Programming Language: Python

VII. CONCLUSION

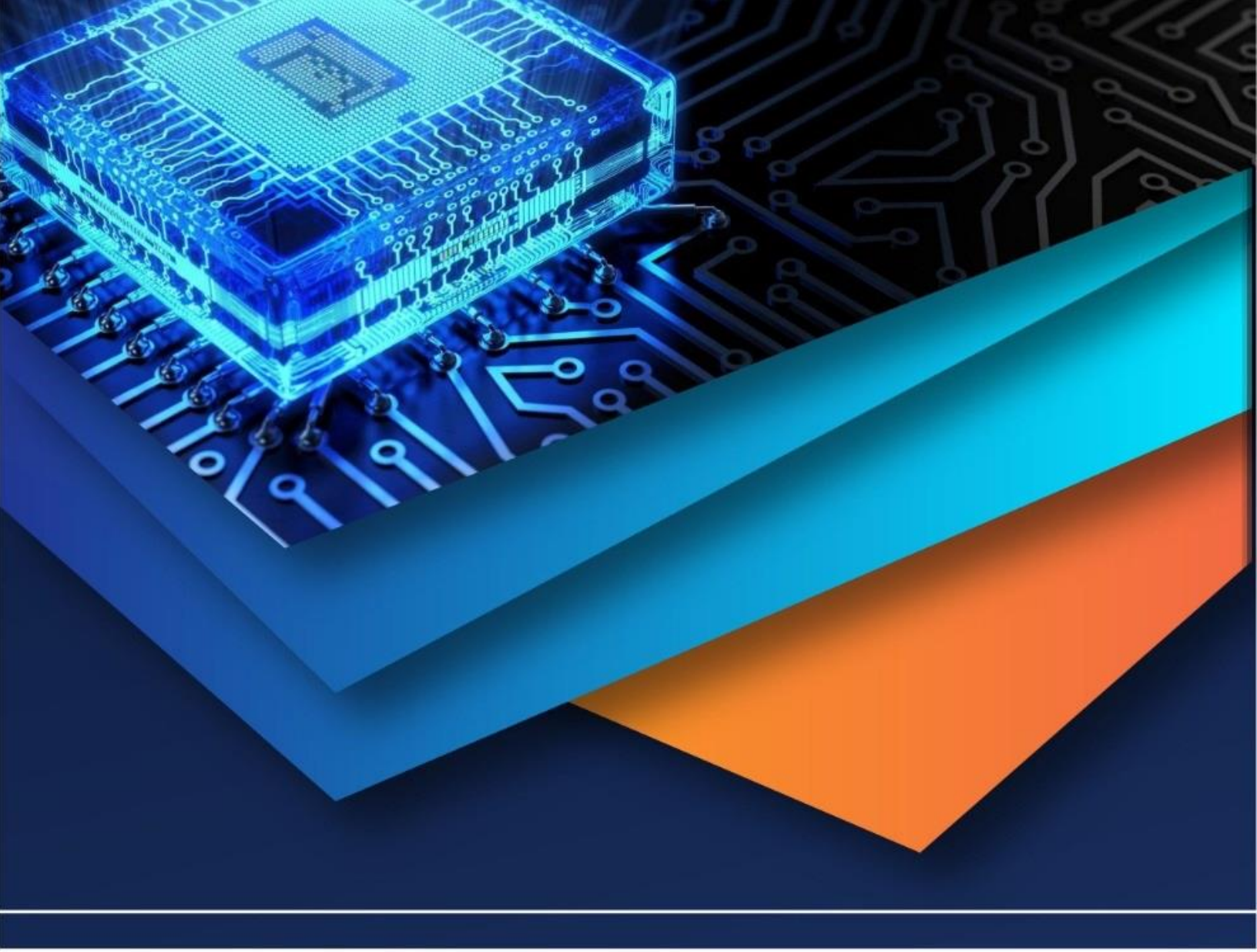
In conclusion, virtual sketching using OpenCV is a powerful tool for creating digital representations of real-life objects and scenes. By leveraging the tools and functions provided by OpenCV, developers can create highly detailed and accurate sketches that can be used for a wide range of applications, from artistic expression to design and education. The objective of making a virtual sketch using OpenCV is to create a digital representation that is precise and controllable, allowing for experimentation and iteration without the constraints of physical mediums. Additionally, virtual sketches can be easily shared, modified, and edited, making it easy to collaborate and receive feedback.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The purpose of making a virtual sketch using OpenCV can vary depending on the specific application, but common purposes include artistic expression, design, animation, visual effects, and education. Virtual sketching allows artists, designers, and educators to experiment with different styles, techniques, and materials in a digital environment, providing a level of precision and control that traditional methods may not offer. Overall, virtual sketching using OpenCV is a powerful tool that enables developers to create highly detailed and accurate digital sketches for a variety of applications. With its wide range of tools and functions, OpenCV provides developers with the ability to create complex and visually stunning sketches that are both beautiful and functional.

REFERENCES

[1] Tessier, A., Lindeman, R., & Bowman, D. A. (2016). Virtual Sketching: A System for Real-time Freeform Modeling in Virtual Reality. In Proceedings of the 2016 Virtual Reality International Conference (VRIC).
[2] Tessier, A., Lindeman, R., & Bowman, D. A. (2016). Virtual Sketching: Extending 3D Sketching to Mixed Reality Environments. In Proceedings of the 2016 IEEE Symposium on 3D User Interfaces (3DUI).
[3] Long, F., Zhang, X., Chen, Y., Zhang, J., & Chen, X. (2017). Sketch-and-scan: A virtual sketching system for 3D reconstruction. IEEE Transactions on Visualization and Computer Graphics, 23(5), 1425-1438.
[4] Lai, Y., & Lin, W. Y. (2017). Real-time virtual sketching with depth-adaptive stylization. Computers & Graphics, 64, 33-43.
[5] Steenbakkers, J., & van Toll, W. (2017). Virtual Sketching of 3D Objects. & Kim, J. (2018). A Virtual Sketching System for 3D Character Design. International Journal of Human-Computer Interaction, 34(9), 870-880.
[6] Bertini, E., Brusati, G., & Mozzetti, F. (2018). Virtual Sketching System for 3D Printing: A Preliminary Study. In Proceedings of the 2018 International Conference on 3D Vision (3DV).
[7] Aslan, I., & Erdem, A. (2018). Virtual Sketching: Augmenting Sketching with 3D Modeling Tools in Virtual Reality. In Proceedings of the 2018 IEEE Symposium on 3D User Interfaces (3DUI).
[8] Yang, Q., Xu, C., Wang, Y., & Zhang, W. (2019). Real-Time Virtual Sketching for Interactive 3D Character Design. Journal of Computer Science and Technology, 34(2), 242-256.
[9] Prasad, V., & Nair, N. (2019). Virtual Sketching of 3D Objects using Deep Learning. In Proceedings of the 2019 International Conference on Signal Processing and Communications (SPCOM



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)