# **Julia in Computer Vision**

# Varun Paul Reddy Gade (U00342648) Arun Bharathapu (U00347356)

**Abstract:**

In this project an attempt is made to bring out the capabilities of Julia for computer vision in face detection and face recognition. The Julia programming language, with its unique combination of ease of use, expressiveness, and computational efficiency, has attracted significant attention in the computer vision community, offering a compelling alternative to traditional languages like Python and C++ for implementing computer vision techniques. In the context of surveillance systems, where face recognition and detection play crucial roles, Julia's capabilities make it an attractive choice for researchers, developers, and practitioners. This project provides a comprehensive overview of computer vision techniques for face recognition and detection using Julia, focusing on surveillance systems and comparing its performance to Python.

**Index Terms:**

1. What is the point of Julia?
2. Computer Vision: What it is and Why it matters
3. Julia in Computer vision
4. Conclusion

# **What is the point of Julia**

Julia is referred to as a dynamic, general-purpose, high-level, high-performance language. Julia's development had as one of its primary goals the creation of a language that would allow programmers to produce fast, low-level machine code that is typically found in static languages while still allowing for clear, high-level, generic, and abstract code resembling mathematical formulas. [2]

To solve problems related to big data and analytics, Julia combines the speed of production programming languages like Java and C++ with the functionality of quantitative environments like R and Python.[1]

The four youngsters at MIT who created Julia needed to address the weaknesses in Python different dialects, and applications utilized for logical figuring and information handling. They desired:

* Language that is freely available as open source and licensed.
* A homoiconic language, with genuine macros like Stutter, however with a self-evident, natural numerical documentation like Matlab.

Something that is as useful for general programming as Python, as simple for statistics as R, [2] as natural for processing strings as Perl, as powerful for linear algebra as Matlab, and as adept at joining programs as the shell.

An easy-to-learn feature that challenges even the most committed hackers. Intuitive and effortlessly incorporated.

**Use cases of Julia programming**

* Machine Learning/ AI
* Data science and visualization
* Web development
* Graphs
* Parallel computing
* Robotics
* Scientific computing
* Audio development
* Game development

**Features of Julia**

Julia programming Language main features are

* Multiple dispatch
* Designed for distributed computing
* Like C it performance will approach
* Without wrappers or special APIs it call C function directly.
* It generates automatic code for different argument types.

# **Computer Vision: What it is and why it matters.**

Computer vision concepts for face recognition and detection techniques, and the role of computer vision in surveillance systems.

Computer vision is one of the areas of artificial intelligence that trains and enables computers to comprehend the visual world. Digital images and deep learning models enable computers to precisely classify, identify, and respond to objects. [3]

Computer vision in Artificial intelligence is committed to the advancement of computerized frameworks that can decipher visual information, (for example, photos or movies) in similar way as individuals do. The teaching of computers to comprehend and interpret images pixel by pixel is the objective of computer vision. The field of computer vision is built on this. Concerning specialized side of things, computers will look to extract visual information, oversee it, and break down the results utilizing complex software programs. [5]

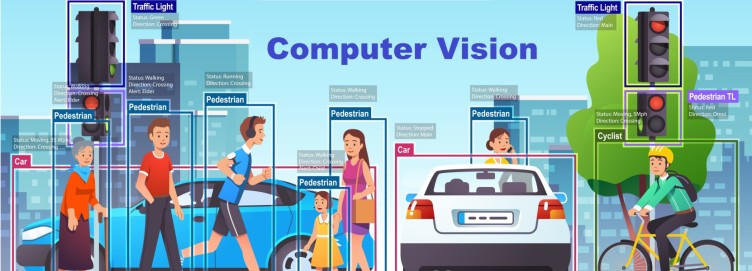


Figure: computer vision Adds value to business [3]

**Computer Vision Applications**

The applications that included in the computer vision are the following: [5]

* **Facial Recognition:** This area of research is heavily utilized in facial recognition software, which makes use of computer vision to identify individuals in photographs. Facial attributes in photographs are recognized by computer vision algorithms, which then, at that point, match those perspectives to put away face profiles. Face recognition is increasingly being used to verify the identity of people using consumer electronics. Social networking applications employ facial recognition for user tagging and detection. Using surveillance footage, law enforcement uses face recognition software to identify criminals for the same reason.
* **Categorization of Images:** A computer program that makes use of image categorization has the ability to identify what an image is a dog, a banana, a human face, etc. In particular, it may assert with conviction that an input picture falls into a particular category. It could be used by a social networking platform, for example, to filter out photos that people post that they find offensive.
* **Object Detection:** Object detection may then use information to search for and catalog instances of the desired class of images by first categorizing them. In the assembling business, this can remember finding defects for the creation line or finding broken equipment.

**Computer vision Algorithms**

The various techniques used to understand the objects in digital images and extract high-dimensional data from the real world to produce numerical or symbolic information are included in computer vision algorithms. There are numerous other computer vision algorithms engaged with perceiving things in photos. [2] [1]

Common examples include:

* Object Classification
* Object Identification
* Object Detection
* Object Segmentation
* Object Verification
* Object Recognition
* Object Landmark Detection

**Advantages and Disadvantages of Computer Vision**

Let’s discuss about the Advantages and Disadvantages [6] of Computer Vision

**Advantages:**

* Process will be simple and fast: It permits industries and customers to check. Additionally, it grants them access to their goods. Computer Vision is present in fast computers, making it possible.
* Reliability: Cameras and computers lack the human element of fatigue, which is eliminated in them. Most of the time, the efficiency is the same, and it doesn't depend on things like illness or sentimental status.
* Accuracy: the accuracy of compute Envisioning, and Computer Vision will guarantee a superior exactness on the product outcome.
* Numerous applications: The same computer system can be seen in numerous activities and fields. Additionally, through scanned images in the medical field and factories with warehouse tracking and supply shipping, among other diverse options.
* The decrease in expenses: Computer Imagining cuts down on both time and the number of errors that are made. It saves money on hiring and training special employees to perform the same tasks as hundreds of workers using computers.

**Disadvantages:**

* Expertise is required: Specialists in the fields of machine learning and artificial intelligence are in high demand. A professional who can make full use of Computer Vision and understands how those devices work.
* Spoiling: In some situations, removing the human element may be beneficial. However, the machine or device does not notify users or anticipate problems when they occur. In contrast, a human being is able to anticipate when another person will not show up.
* Inability to process images: Computer Vision and image processing are highly likely to fail if the device malfunctions due to a virus or other software issues. However, if we fail to resolve the issue, the device's functions may vanish. In warehouses, it may freeze the entire production. [6]

**About Face detection and Face recognition**

Face detection and face recognition are two related but distinct technologies used in computer vision and image processing. [2]

Face detection is the process of identifying human faces in images or videos. This is usually done by applying image analysis and machine learning algorithms to identify patterns and features that are typically found in human faces, such as the position and shape of the eyes, nose, mouth, and jawline. Face detection is often used in applications such as video surveillance, biometric security systems, and social media.

Face recognition is the process of identifying a specific individual based on their facial features. This is usually done by comparing the detected face to a database of known faces and attempting to match it to a specific individual. Face recognition is typically more complex than face detection, as it requires more advanced algorithms that can handle variations in lighting, pose, and facial expressions. Face recognition is used in a variety of applications, including access control, law enforcement, and social media.[5]

# **Julia in Computer vision**

Julia is a high-level programming language designed for numerical and scientific computing, including applications in computer vision. Julia provides a flexible and high-performance environment for implementing computer vision algorithms, making it a popular choice for researchers and developers in the field. [4]

And also present an overview of Julia, highlighting its advantages for computer vision tasks, such as exceptional performance, ease of use, and a growing ecosystem of libraries and tools that cater specifically to computer vision applications.

A machine learning library is called Flux. It comes "batteries-included" with numerous valuable devices worked in, yet in addition allows you to utilize the full force of the Julia language where you want it. We follow a couple of key standards: [7]

Doing what is obvious. There aren't many explicit APIs for Flux. In its place, the mathematical form can be quickly written down.

The default is extensible. Flux is written to be extremely adaptable and fast. Broadening Motion is basically as straightforward as involving your own code as a feature of the model you need - it is all undeniable level Julia code.

Be respectful of others. Instead of duplicating Julia libraries, Flux works well with them, from differential equation solvers to images.

**Differences between Julia and Python**

The following is a comparison of Python and Julia.[8]

* **Popularity**

As of late, Python has become one of the most famous programming language. It has a developer community that is among the largest of any language, and it is more than 30 years old. The Python programming language offers arrangements and backing for each likely issue. It still has a small but dedicated fan base, and the writers provide the most support, despite Julia's constant growth. How to use Julia across multiple platforms is the subject of blogs and a growing community. Julia's popularity will ascend as it extends past Information Science. Frameworks for web development are now available to language developers. Due to the expansion of development options, there will be more developers using it.

* **Speed**

When writing code, speed of execution is critical. The Julia programming language executes at a comparable speed to C. Julia was made as a fast-programming language. There is no interpreted mode for the Julia programming language that can speed up execution. The LLVM framework is used in the creation of Julia's programs. Big Data, cloud computing, and data analysis all require solutions that can be found in the Julia programming language. Julia's performance is quicker and more effective than that of Python.

* **Libraries**

Python comes with a powerful library that makes coding in Python easier. As functions, Python code can import and utilize such libraries. There are not many libraries in Julia, which places it in a difficult spot contrasted with Python. Additionally, Python is supported by many third-party libraries. Additionally, Julia's libraries suffer from inadequate package maintenance. While plotting information interestingly, Julia can speak with C libraries.

* **Dynamically Typed**

Julia and Python both use dynamic typing. Variables can be used in code without being explicitly declared. Additionally, Julia is a dynamic as well as a static language. Depending on their requirements, coders can use it either way. Julia lacks this advantage over Python.

* **Parallelism**

Julia and Python both support simultaneous operations. Serializing and desterilizing information across strings is essential for Python methods. Julia, on the other hand, employs more advanced parallel techniques. Also, Julia's parallelization language structure is less unbalanced than Python's, decreasing its pertinence.

* **Tooling support**

Programmers favour languages with excellent support and tools. Python's tool support is superior to Julia's. Python, in contrast to Julia, offers superior tooling support. Subsequently, Julia's exhibition instruments are less broad than those of Python. Additionally, Julia is more susceptible to unsafe interfaces due to its native

* **Versatility**

Python is a versatile language because it makes coding and reading simple. Because of its adaptability, Python is an excellent platform for developing, automating, and scripting websites. Python is preferred by developers due to its speed of execution. A few libraries and structures are utilized to diminish improvement time. When confronted with difficulties in scientific programming, Python is more adaptable than Julia.

# **Code implementation and Results**

The packages are:

**PyCall**

The Julia language can be directly called by this package, allowing for full Python interoperability. [9]Large data structures can share between Julia and Python without copying them, define Python classes from Julia methods, call Python functions (with automatic type conversion between Julia and Python), and import any Python module from Julia.

Additionally, the PyCall module provides the brand-new type PyObject, which is a wrapper for PyObject\* in the C API of Python and represents a reference to a Python object. [9]

Presently, the sorts upheld are numbers (number, genuine, and complex), Booleans, strings, IO streams, dates/periods, and capabilities, alongside tuples and clusters/records thereof, yet more are arranged. (Python strings are created from Julia symbols.)

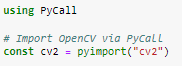
**ImageView**

ImageView.jl gives capabilities to see pictures in discrete windows. To find out about what ImageView.jl is able to do, it is valuable to test the packages after installation. [1]

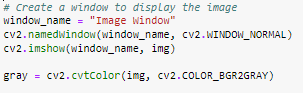
**Images**

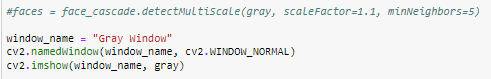
Images.jl is becoming more and more of an "umbrella package," exporting a collection of packages that can be used for common image processing tasks. Julia Images, Julia Arrays, JuliaIO, Julia Graphics, and Julia Math host the majority of these packages. [1]





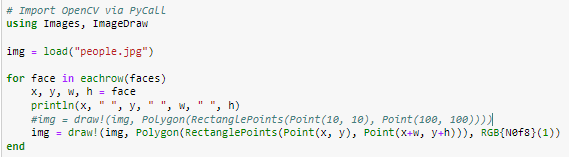
D:\shalu projects\FINAL DOCUMENTS OF MINI PROJECTS\computer vision\Capture 3.PNG



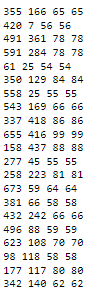


D:\shalu projects\FINAL DOCUMENTS OF MINI PROJECTS\computer vision\Capture 6.PNG

D:\shalu projects\FINAL DOCUMENTS OF MINI PROJECTS\computer vision\Capture 7.PNG



**Output:**



**Result:**

D:\shalu projects\FINAL DOCUMENTS OF MINI PROJECTS\computer vision\Capture 10.PNG





# **Conclusion**

Overall, this project offers a comprehensive understanding of computer vision techniques for face recognition and detection using Julia, focusing on its application in surveillance systems. By comparing Julia and Python's performance and emphasizing Julia's unique benefits for computer vision tasks.

References:

1. Julia, “Julia 1.8 Documentation”, <https://docs.julialang.org/en/v1/>
2. Nihal Singh “Introduction to Deep Learning in Julia”, <https://www.analyticsvidhya.com/blog/2021/10/introduction-to-deep-learning-in-julia/>
3. Steve Nouri,” How Computer Vision Adds value to Businesses”, <https://www.linkedin.com/pulse/how-computer-vision-adds-value-businesses-steve-nouri/>
4. Zacharias Voulgaris, “Julia and Its Usefulness in Data Science”, <https://towardsdatascience.com/julia-and-its-usefulness-in-data-science-6e9181504ff4>
5. By Simplilearn,”What Is Computer Vision: Applications, Benefits and How to Learn It” <https://www.simplilearn.com/computer-vision-article>
6. “ADVANTAGES AND DISADVANTAGES OF COMPUTER VISION”, <https://master-artificialintelligence.com/advantages-disadvantages-computer-vision>
7. Mcabbott,” FluxML/Flux.jl”, <https://github.com/FluxML/Flux.jl>
8. West Agile Labs,” Julia Vs Python : Which Is The Best Programming Language”, <https://www.westagilelabs.com/blog/julia-vs-python/>
9. “PyCall.jl: Call Python Functions from the Julia Language”, <https://morioh.com/p/d4e966c1b0d7>