A Hitachi Group Company

Edge Computing for Facial Recognition & Emotion Detection

by Matt R. Cole,
Manager, Engineering

Contents

Overview

Facial recognition and facial emotion detection are perhaps two of the biggest buzzwords in use today. We've all seen and used facial detection, the simplest example being the facial recognition used to gain access to your mobile phone. In many other cases, facial recognition is being used without your awareness, such as in heavily crowded public places such as Times Square in NYC.

Its usage can vary heavily depending on the country, as well. With the advent of 5G, facial recognition moved out to the Edge, which will become more and more prevalent.

This paper will discuss the major players in both spaces, some key features you should be aware of, and how this biometric technology can operate both in the cloud and at the Edge. We will also dive into some of the many decision factors an implementer needs to consider, such as hardware, software, and cost of ownership.

We will explore the fascinating world of facial emotion recognition and here, too, we will discuss the various hardware and software considerations you need to keep in mind. You will find various use cases for studying. Finally, you'll find recommended courses and additional reading to continue studying this fascinating technology further.

Last but definitely not least, we will discuss Edge computing in detail.

Edge Computing: An Introduction

Edge Computing describes a distributed framework that is designed to move computing resources closer to the applications that use them by using co-located data servers.

When we talk about Edge Computing, we are no longer talking about single endpoints but rather describing 'layers' of data with closer proximity to the application than typically seen in a data center. If your application or business requires more reliable, sub-second response times, 'Edge Computing' may just be the answer!

When we discuss Edge Computing, there are two models of operation we are concerned with, 'stand-alone' and 'distributed.'

With a stand-alone model, you typically have some kind of a smart device, for instance, a smartphone. This device is co-located (or closer) to the application's end-user. The application (yours) is always available because it is located on the processing device. Additionally, data is also available on the device, so the device itself now becomes the data center. The data may later be synced with other devices or data centers, but the main processing component is now directly linked with the end-user. Network latency in this respect has now been removed from the equation as all the required data is right there on the device.

With a distributed model, we are now dealing with smaller data centers, such as IT Closets (think Call Centre, server racks in big-box retailers, and so on). With a distributed model, your footprint is determined on the scale and the total number of downstream edge or IoT end-points you have.

These 'micro-data-centers' provide the power to the distributed model. They allow your business to benefit from Edge Computing while at the same time bringing compute and storage resources on-prem, which will reduce downtime and application responsiveness. A good example of this might be a large-scale AI application, which can do some processing locally but needs the larger compute resources to provide additional benefit.

Many people are now trending towards Edge Computing, which represents the best practice for tomorrow's applications. They do so because of the benefits provided by Edge Computing, which are:

Speed

Although Edge Computing provides many benefits, there are times when the data is too far from the processing application to provide sub-second response times. If your application requires this, then the proper architecture and approach is a must.

Sometimes, you simply cannot achieve this type of speed by optimizing your network due to cost and technology standpoints. If you look at specific types of applications, such as Health Care, they usually rely on high-volume, low-data, in-transit data types. They have very low latency, as in sub-milliseconds, placing critical importance on access times. Even if you had the best internet bandwidth, this type of application cannot usually be served with the conventional cloud-only approach. You have to move computing to the Edge to gain the necessary speed requirements.

Security

In today's world, topics such as data privacy, governance, and regulatory policy are all things we need to think about. These items are required in almost all applications that handle sensitive data. If you look at a domain such as Health Care, regulations typically require that sensitive information never leave the premises.

There are other domains as well in which data privacy is paramount. If you look at the healthcare devices out in the market, such as FitBit and other fitness trackers, many of which provide recommendations based on lifestyle, location, etc. This information must be processed locally because consumers simply do not want it stored or processed in the cloud. If you step back and look at all the data-privacy intrusion attacks, such as the famous 'Target' data breach, you can understand why.

Whether you have your data processed on the device itself or a micro-service co-located out of the cloud and closer to the device, this micro-tier processing model can meet the requirements and regulations, as well as address data breach concerns.

Bandwidth

Cloud computing itself offers many advantages and savings in terms of efficiency. Higher bandwidth is usually one of those, but under certain conditions, it may not be. Moving the compute resources to the Edge will create additional savings by reducing bandwidth, reducing load, lowering the volume of transferred data, and more. Doing so also makes predicting costs more reliable because, in part, you can better estimate bandwidth requirements based upon the Edge device distributions and expected throughput.

Reliability/Resilience

When you move resources to the Edge, you automatically guarantee business uptime because the compute resources have uninterrupted access to the data it needs to operate. The beauty of this is that this can happen even if the internet connection is interrupted. The better your application manages interrupted connections, the better experience your users will have. This, in turn, affects measurable metrics such as experience, repeat business, loyalty, and more.

Internet connectivity, by default, introduces a single point of failure by itself, and sometimes there is little control over predicting or managing this. Obviously, as a business, this is the last thing you need. Even though Edge Computing still involves syncing your data across the internet to the cloud for eventual storage, layering compute layers for data acquisition and processing closer to the devices and apps via microdata centers speeds up processing and eliminates single points of failure.

One final point on Edge Computing. When appropriately implemented, this will require newer development techniques for applications, which now need to prepare themselves as an application that will never go down – even while increasing processing speed. In many cases, our applications have been developed with connectivity problems, handling slower data speeds, etc. We will now have light-weight data storage directly in our applications, which also implies built-in data synchronizing to ensure the data is always up-to-date and not corrupted, etc.

In some cases, this will make the app development process faster and less expensive. Developers no longer need to focus on speed and connectivity issues but rather the core competency of the application.

Facial Recognition

Facial Recognition is a technology that many of us use regularly. If you own an iPhone or an Android-based smartphone, chances are you are using facial recognition to gain access to your phone. Although becoming mainstream only in the past several years, facial recognition itself has been around for some time. Let's start by defining what facial recognition is.

Facial Recognition is the process of identifying or verifying (important distinction here) a person's identity using a digital image of their face. This technology scans, captures, analyzes, and compares different patterns based on specific facial details.

The Facial Recognition process consists of:

1. **Facial Detection**: Location of human faces in images and videos.
2. **Face Analysis**: Scan, capture, and analyze an image or video into a set of digital information based upon a person's features.
3. **Facial Identity**: Verify whether the facial image belongs to the person claiming ownership.

Facial recognition is a form of biometric security that analyzes various facial vectors while comparing them to images that have been 'pre-enrolled.' Once you have an image of an individual (or cat or dog, for that matter), you can then pick that person out of a host of many.

It is important to clearly articulate the difference between "identify" and "validate".

Identify answers the question: "*Who are you?*"
Validation answers the question: "*Are you really who you say you are?*"

So why should we use facial recognition as a part of our biometric process? Quite simply because:

1. Facial Recognition has become the preferred biometric security benchmark.
2. Facial Recognition is easy to deploy and implement.
3. Facial Recognition requires no physical interaction with end-users.
4. Facial Recognition performs identification and verification in a speedy manner.

Each individual is unique and has facial features and variables that are unique to them. Among them are:

- Depth and width of your nose
- Forehead length
- Shape of eyes
- Cheekbone style
- Chin size and shape
- And so on.

Facial recognition uses various algorithms and mathematical formulas to analyze this information and connect it to an individual image. Once scanned and analyzed, this information is usually stored in template form as a part of an 'enrolment' process. Once this has been completed, the template can be used any time when identification is required.

Several years ago, we went from requiring strict, frontal images to now utilizing the deep neural networks capable of leveraging thousands of data points from many different angles.

We went from very rudimentary products all the way to products such as FaceMe1, which is now recognized as the world's leading AI facial recognition engine. Such algorithmic products have become so far advanced they can identify the same person with or without a mask!

Major Players (Pros)

Although there are many players in this space and the list continues to grow, these are some of the major players at the time of publication.

GaussainFace

Although not a 'player' per se, this algorithm is sometimes attributed to starting the recent rapid progression of facial identification scores, comparing computers and humans. This algorithm was developed back in 2014 by researchers at The Chinese University of Hong Kong, who achieved facial identification scores of 98.52% (compared to a score of 97.53% achieved by humans).

Facebook

Facebook and Google are two of the larger contributors in this space, and both have made significant contributions over the past few years. In 2014, Facebook announced 'DeepFace,' a product capable of determining whether two photographed faces belong to the same person. The accuracy rate achieved for this was an impressive 97.25%. When accomplishing the same thing, humans achieved a score of 97.53%, only mildly better than Facebook.

Google

As usual, Google upped the stakes in June 2015 when it introduced 'FaceNet'. When tested in the Labelled Faces in the Wild (LFW) dataset, FaceNet achieved a record score of 99.63% (+- 0.0009). This technology was subsequently integrated with Google Photos and used to sort pictures and automatically tag them based upon recognition.

Amazon

As you would expect, Amazon is a major player in this space. With Rekognition, Amazon allows you to provide visual recognition capabilities at scale. Rekognition is said to recognize as many as one hundred people within a single image and perform face matching against databases containing tens of millions of facial images.

The usage of Amazon CloudFormation is an Infrastructure as Code solution that helps facilitate the process of building the models then deploying them throughout their lifecycle.

AWS MarketPlace also offers a fascinating product, Deep Face Recognition API, a toolkit for developers who offer facial recognition technology in their applications with minimal effort. Want to see how to turn your city into a Smart City? Just watch this demo.

Finally, for full disclosure, it is appropriate to also mention some of the downsides of this technology.

Major Players (Cons)

Amazon

In a July 2018 article, Newsweek reported that Amazon's facial recognition technology falsely identified twenty-eight US Congress members as having prior criminal records.

Microsoft/IBM

In a study done by MIT researchers in February 2018, researchers found that Microsoft, IBM, and China-based Megvii (FACE++) tools had very high error rates (gender bias) when attempting to identify darker-skinned women compared with lighter-skinned men.

Aside from those specifically mentioned above, there are many other major vendors when it comes to facial recognition:

- Accenture
- Aware
- BioID
- Certibio
- Fujitsu
- Fulcrum Biometrics
- Thales
- HYPR
- Idemia
- Leidos
- M2SYS
- NEC
- Nuance
- Phonexia
- Smilepass

Let's move forward by understanding some key features of facial recognition.

Key Features

Face Detection

This is the first step in the facial recognition process. Typically, the entire face is scanned, and sometimes multiple phases or steps are used to determine the subject being analyzed. In today's ever-changing society, we see facial detection used in places such as public areas (airports, bus stations, train stations, streets, etc.). The need for fast detection has become paramount. Additionally, the need for identifying more than one face at a time – for example identifying multiple people in a crowd – is becoming more and more applicable to high-priority use cases.

Feature Extraction

The second step in the facial recognition process is extracting features from the obtained image(s). During this step, the recognition engine will extract an N-dimensional vector set from the image(s) it obtained.

This information will usually be stored as 'the template.' The higher the 'N' value, the higher the precision. Typically, for facial recognition to be as accurate as possible, it requires very high accuracy.

Recognition

In the final step, after we have detected and processed our image(s) and created our template, this template is then stored, or 'pre-enrolled,' into a storage system such as a database in a centralized location. This will then serve as the master collection of information later used in identifying select individuals.

Although the features above can be considered the key or 'main' features, there could be additional features depending upon the application. Let's talk about a few.

Facial Attribute Detection

This is different from feature extraction as it is designed to specifically identify macro attributes such as gender, head movements, gender, mood, head orientation, and much more. Such attributes are recognized for more targeted purposes such as digital billboards (for example, identifying someone as female, so as they walk towards a digital billboard, the billboard will produce custom ads based on that attribute).

Mask Detection

This is perhaps the latest phase of facial recognition technology and has sparked much interest in our recent pandemic. Mask Detection can determine whether you are wearing a mask, whether it covers your mouth and nose adequately, and more.

Anti-Spoofing

Anti-spoofing is an exciting new technological use for facial recognition. To bypass facial recognition, sometimes individuals attempt to hold someone else's photo (or video) in front of a camera.

This type of intrusion has been attempted so often that the latest facial recognition software can detect 'liveliness' using 2D and 3D cameras. This allows for the detection of head movement as well as depth detection. Of course, such implementations are much more expensive in 3D than in 2D, but it is a requirement for some purposes due to security levels.

Cloud-based Recognition

Facial recognition requires vast amounts of data. A cloud-based implementation over the internet can result in an expensive and limited application. Network availability and bandwidth are always issues. Network congestion can slow down processing due to sub-par speeds, even to the point where performance is completely inadequate.

Additionally, since the images are sent over the internet, we now have security and privacy issues. The place of storage could be vulnerable to hacks or data leaks. Even though the cloud offers many advantages, no solution is perfect all the time. We also need to consider the size and scale of the solution. If you are developing a small-scale doorbell camera system, you may adequately process images and alert customers. But larger, more complicated systems will almost always encounter issues.

The advantage of a cloud deployment is that there is no infrastructure management, assuming your cloud is not on-prem. The reality is that many facial recognition systems have been and are built on and in the cloud, but this is mostly because the proper alternatives like Edge AI chipsets were either unavailable, slow, or unaffordable.

Microsoft Azure Face API, Google Vision AI, and Amazon AWS Rekognition are just a few such products.

Edge-based Facial Recognition

We've talked about cloud-based facial recognition, and now it is time to discuss Edge-Based Facial Recognition.

We should probably start with defining what we mean by 'Edge.' When we use the term 'Edge Computing,' we are referring to moving various functionality out from the central confines of the cloud and towards or at the point of interaction with a user or the intended functionality.

For instance, when dealing with Smart Traffic devices on the Edge, we have moved the algorithms and processing code from the cloud to the actual traffic cameras themselves. Here is an illustration showing this concept:

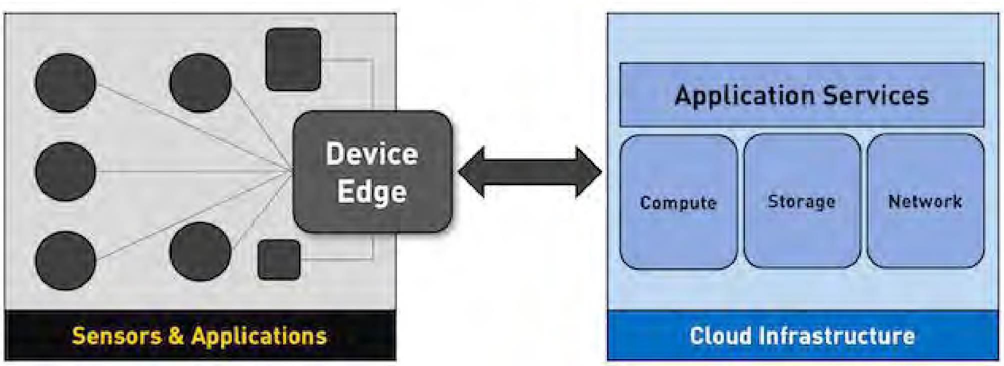

Image Credit: Device Edge - Janakiram MSV

Edge-based facial recognition means that the recognition and processing of images are embedded, and take place in, local devices. The processing happens at the 'edge' of the network. Since these devices are embedded using the correct hardware, they can run fast and with extreme precision because they are not subject to network-related issues and other factors.

The processing itself should be high-speed, and the only time the network or cloud needs to be involved is when it is time to contact the central face database (assuming that it is not also stored on the Edge device, which it may be). Fortunately, if network or cloud connectivity is required, it should be much less than other solutions if compressed and encrypted templates are used. Some examples would be your smart doorbell, smart cameras, digital signage, and more.

Another compelling area Edge could greatly accelerate in is the financial industry. Notorious for ensuring that most of the workforce are in-office versus working remotely (until recently), finance has been very firm on not allowing internet connections in some areas because of security concerns. Having processes located at the Edge alleviates many of the security issues which can arise.

Hardware Considerations

When dealing with Edge Computing and facial recognition, we have to consider the hardware being used. It is imperative to accomplish our mission, and there is no one-size-fits-all solution here. It depends upon your industry, application, and users and should be tailored as such.

One of the most important considerations for hardware is the chipset being used. AI chipsets or ‘System-On-Chip’ (SoC) must have the right computational power, proper form factor, storage, low power consumption, and (perhaps most importantly) the AI inference engine on the chip itself.

Fortunately for us, large manufacturers such as NVIDIA, Intel, Qualcomm, and others recognize the need for this power. The recognition is so evident that new types of chips are being brought to market, such as **APU** (AI Processing Units), **VPU** (Vision Processing Units), **NPU** (Neural Processing Units), and more.

These types of chips can play a significant role in every aspect of the facial recognition process. Let’s examine just a few so that you know some of the many choices you have.

Vendor/Type	Product	Highlights
NVIDIA GPU	T4	NVIDIA T4 is a universal deep learning accelerator ideal for distribution in computing environments. T4 supports all AI frameworks and network types, delivering dramatic performance and efficiency that maximizes the utility of at-scale deployments.
NVIDIA GPU	A40 RTX A6000	These are designed for workstations or on-premise servers, handling vast amounts of facial recognition requests.
NVIDIA GPU	Quadro RTX 5000	Quadro 4000/5000 series are designed for workstations to handle facial recognition requests from small to midsize camera numbers.
Intel VPU	Movidius Myriad X - MA2485	Myriad X VPU can deliver a total performance of over four trillion operations per second (TOPS). Vendors like iEi or Advantech put multiple VPUs into a single AI acceleration card.

Vendor/Type	Product	Highlights
NVIDIA SoC + GPU	Jetson AGX Xavier	This chip provides a strong balance between performance, low power consumption, form factor, and price - is a great option for small to medium-sized workstations.
Intel CPU	Atom x6000E	Facial recognition algorithms that integrate VNNI can significantly boost (two times faster) on the same CPU.
Qualcomm SoC + GPU	QCS610 QCS410 QCS603 APQ8053	On-device machine learning through the Qualcomm AI Engine can support a plethora of AI networks and IoT use cases with low power consumption. It is purpose-built to deliver high-performing, power-efficient Edge Computing for next-gen smart cameras, smart enterprises, home, and automotive IoT applications.
Rockchip SoC + GPU	RK3399 Pro	The RK3399Pro provides super performance and a one-stop turnkey solution for AI.
Google SoC + GPU + TPU	Coral Edge TPU (SOM)	The Coral SoM is a fully-integrated Linux system that includes NXP's iMX8M system-on-chip (SoC), eMMC memory, LPDDR4 RAM, Wi-Fi, Bluetooth, and the Edge TPU coprocessor for ML acceleration. The onboard Edge TPU coprocessor is capable of performing four trillion operations (tera-operations) per second (TOPS), using 0.5 watts for each TOPS (two TOPS per watt).

Now that we've discussed some of the chipsets available for facial recognition, it would seem only appropriate to talk about the different Operating Systems (OS) that run on these chips. We can summarize the availability of the operating systems under four main systems, with Linux having many variants.

Here is a brief list:

- Windows
- Android
- iOS
- Linux
    - Ubuntu x64
    - Ubuntu ARM
    - RedHat
    - JetPack
    - CentOS
    - Yocto ARM

System Architectural Considerations

When dealing with any application, especially facial and emotion recognition, one must pay attention to ensure the proper architecture is in place. There is a great deal of flexibility and latitude in many applications because they are dealing with CPU-based solutions only, or an occasional TPU-GPU here and there. But when we are dealing with Edge Computing, the hardware becomes even more critical to ensure we are doing it right.

We have many choices for things such as GPU/TPU/VPU. The cards and chips can range from basic to advanced processing capabilities and power. Additionally, when designing a facial or emotional recognition system, one must also consider that there will more than likely be many concurrent video streams running over the system between the processors. If care is not taken towards the architecture, your final implementation may be too slow to be useful.

One key action would be to minimize the data flow and traffic between the CPU/GPU/TPU/VPU and memory. To give you a better sense of what you are dealing with, it is very feasible that an adequately designed facial or emotional recognition system will handle between twenty to fifty concurrent video channels. If you then consider that there are usually ten frames per second for each, this can break down into roughly two hundred to five hundred frames per second. The exact number can vary based upon several factors. This should give you an idea of the concurrency level you may be dealing with.

The next thing we must consider when it comes to system architecture is the AI Engine/Model used for facial recognition. Outside of the concurrency issues described above, this may be the single most crucial area of consideration you will face.

Each type of implementation for a facial recognition system will have its own demands and goals, but in several instances, a lightweight AI Model may work just fine. For example, a smart door lock or doorbell will have different requirements and demands than a large-scale identity system that scans everyone in an airport to identify individuals on a no-fly list.

Our next consideration must be the remaining hardware itself. Some applications may be very specific in terms of the hardware type they use. Others may be much more flexible in adaptability across different hardware types. Let's explore this further by identifying some particular market segment implementations of facial recognition systems and addressing each specifically. In this regard, we will be addressing specifics related to facial recognition and not Edge Computing itself, although they may overlap depending upon the scenario.

Facial Recognition: Workstations

When developing and deploying a facial recognition system, you have to address the demand put on the service bus by video channel traffic (see above). Suppose your facial recognition application is a large-scale implementation requiring tens or hundreds of video channels, such as department stores, airports, hospitals, and so on. In that case, your application will benefit from a high-end workstation using the latest GPUs.

This type of implementation may make the most sense to have your smart cameras and devices connected to one or more centralized workstations running the recognition software. This will probably be the most reliable and most cost-effective solution in most cases.

Facial Recognition: Personal Computers

If you have a scenario similar to the above but are not operating on a large scale, you may get by with a PC-based solution. Such examples might be a restaurant or boutique store, etc. In this case, you can probably install one or more devices such as cameras and connect them to a single PC that runs your facial recognition software. Of course, an investment into the capabilities to use GPUs rather than CPUs will pay for itself many times over, but this is isolated to a single computer.

Facial Recognition: Mobile Devices

Perhaps the most recognized application of facial recognition software usage today is via our smartphones and other mobile devices. Be it software to allow access to different data or applications automatically, or using the latest pandemic software, this form of application can in many instances be implemented on a mobile device.

More advanced application forms can also be mobile-based, such as customer identification software, which allows a hotel to automatically recognize that you have entered their premises and automatically check you into your room. Such geo-spacing applications combined with facial recognition are becoming more and more prevalent in places like the world-famous Las Vegas.

Facial Recognition: IoT/Edge Devices

The following scenario is the actual facial recognition on the Edge devices themselves. This type of innovation continues to drive performance while simultaneously cutting costs, and the future looks promising. Global entry travel kiosks, large hotel chains, and more are examples of such implementations. Your Edge device will have something similar to an ARM-based AIoT chip in this scenario. These devices will be connected to the network/internet to access the central storage containing all the pre-enrolled biometric templates. These chips will run the actual AI algorithms used for your processing.

These can either be what is known as a 'System On A Chip (SOC)' or a 'System On A Module (SOM).' These will then connect to a CPU/GPU for processing and acceleration capabilities. In many cases of today's technology, vendors are also including dedicated AI Processing Units (APU) and/or Neural Processing. Units (NPU) like SOC or SOM will allow your application to actually run on the Edge instead of inside the cloud. Power consumption is also a consideration when implementing such solutions. Fortunately for us, most ARM-based devices are very energy efficient and use much less power than a similar PC or Workstation implementation.

Suppose your application is destined for what is called 'thin Edge-based devices,' such as kiosks, tablets, digital signage, other mobile and wearable devices. In that case, this will be a component you will need to ensure it handles what you want to accomplish.

Emotion Recognition

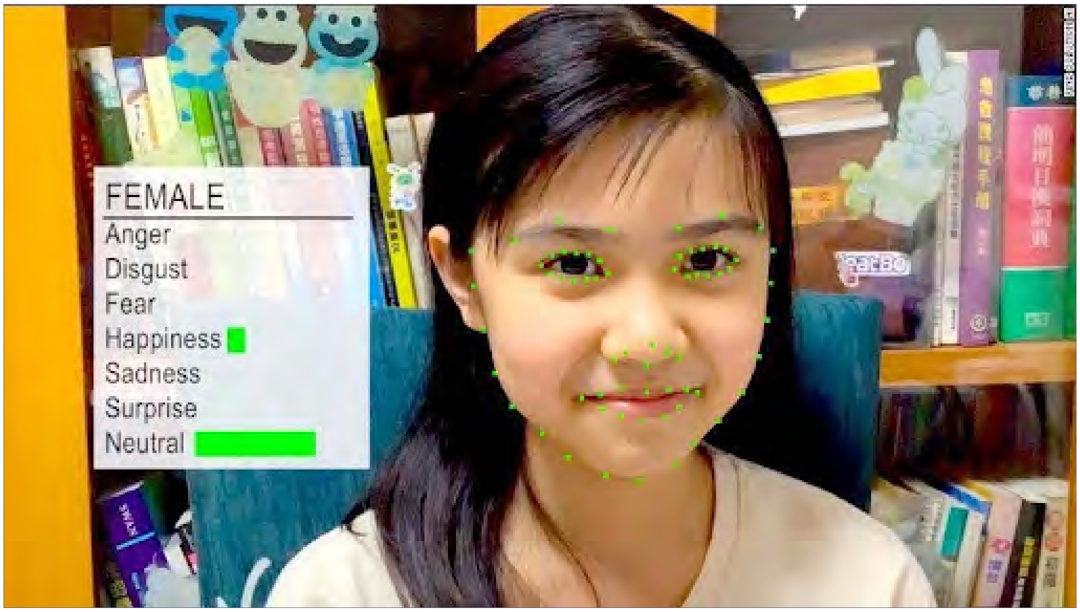

Image Credit: [CNN]

Emotion detection and analysis is a fascinating field of research and provides many opportunities. Actually, this can be referred to by several different terms, such as 'Emotion Detection,' Emotion AI,' 'Aspect-based Sentiment Analysis,' and 'Effective Computing,' just to name a few. Before we go further, let's quickly identify what Emotion AI means in the context of this paper.

Emotion AI, or Facial Emotion Recognition, is simply the process of mapping an individual's facial expressions to identify emotions, such as:

- Disgust
- Anger
- Joy
- Anger
- Fear
- Surprise
- and more.

To accomplish Facial Emotion Recognition, there are three basic steps to follow:

1. Facial Detection (identify the person).
2. Facial Expression Detection (identify the emotion).
3. Assignment of Expression to a Specific Emotional State.

Keep in mind that Emotion AI is the field we are discussing now; it is not limited to facial expressions. It includes facial, voice, text, sentiment, and possible combinations of two or more. Our facial expressions rarely, if ever, tell the whole story.

Seth Grimes, a Natural Language consultant and creator of the Emotion AI Conference, shares this appropriate example:

"If I show you a picture of Kobe Bryant, on the basketball court, smiling, how does that make you feel? Well, Kobe Bryant died a year ago in a tragic crash. So [just] because Kobe Bryant is smiling, that doesn't make you happy. Maybe it's sad, but sadness about someone who has died is actually a positive sentiment. There's a lot of complexity and subjectivity."

You also have to consider the additional complexity of any singular emotion – that there could be different levels of emotion. For example, being 'furious' is more intense than being 'angry,' so varying levels of complexity also need to be factored in.

In a way, this is similar to the concepts we use in Deep Reinforcement Learning, specifically good reward creation. Creating rewards is easy but creating 'good' rewards is quite difficult. When evaluating our goals, we have to evaluate and calculate an appropriate reward for the main goal, but we also have to consider how 'close' the engine got to the goal.

For instance, if we're dealing with being 'angry,' the reward calculation for being 'furious' may be given a greater reward than being 'frustrated.' This means that the proper calculation of the reward has many factors, and for the engine to learn what we really want it to learn, we have to understand and learn how to generate the proper reward. The same applies to emotions. The calculations involved in accurately determining a particular emotion may be complex and subject to scrutiny.

When dealing with emotion, there is also an aspect known as 'Aspect-based Sentiment Analysis.' For example, let's say that someone left a negative review for a recent vacation they took. But what is it that they didn't like? Was it the room, the food, the service, and so on? More concretely, this is the difference between 'imprecise' and 'precise,' which must be considered.

And finally, not everything is black and white when it comes to emotion. Sometimes you deal with what could be called 'narrowly focused' emotion, meaning it's neither precise nor imprecise.

As you can appreciate, accurately determining emotion and sentiment may not always be as straightforward as it seems. How do implementers deal with this, you might wonder? Typically, the solution is to use models that are industry-trained specific rather than more general.

Next, did you know that we can gauge a lot of sentiment via voice analysis? In analyzing a person's voice, there are over a hundred and thirty different characteristics we can gather, all with a great deal of accuracy. Additionally, this part of the market has really put a lot of effort into low-latency coding because they are dealing with real-time voice streams in some cases.

Why does that matter? Because both voice and facial recognition are used to gain a higher accuracy of emotion. The problem here is that NLP, for the most part, is a newer endeavor that has not had the amount of low-latency coding focus that voice has. So, when you combine the two, you find that the processing time is usually longer for facial and NLP recognition than it is for voice, and your overall processing time increases.

What about skin temperature? Recent camera technology will allow detailed analysis of skin pixels, which can also be a factor in determining precise emotion. To do this, you will need a significant investment in the right kind of camera – one with high-quality sensors and very low noise.

Algorithms exist to extract respiration rates from a pulse signal which can only be obtained by analyzing skin pixels. Sounds cool, huh? It's a bit more convoluted than that. For instance, what if the person or the people you are trying to analyze are moving, and therefore, subject to varying light sources or conditions?

People will also have different skin pigmentation, facial hair, and so on. Deep learning usually needs to be employed to increase your precision, which also has a counter effect on the processing time, amount of training data required, the specific focus of models (models for different ethnicities, etc.).

Here's an interesting point you may not have considered. When dealing with facial analysis and emotion, data training is essential. In the instance of emotion itself, people have varying degrees of expressiveness. If you were going to be sure that you had the most precise models for John or Sally, you would train large amounts of specific data to each. Obviously, this is prohibitive, so how do implementers handle this? Simply by averaging.

Typically, an implementer will scan multiple faces simultaneously, extract the information and average it. As you can see, this is far from precise, but the alternative to increasing precision is also difficult or impossible to achieve.

Data acquisition is also a unique challenge when it comes to emotion. As we discussed earlier, you cannot simply make a model based upon every person on the planet. You use a technique known as 'averaging,' as we explained. However, you also have to be aware of introducing 'bias' into your data and ensuring that you don't have over or under-sampling.

The sources of data have to vary. You do not want to take all your images from one source, at the same time of day, for the same ethnicity, and so on. So, where do implementers find such copious amounts of data? By extracting them from public sources from the internet, mostly. Instagram, Pinterest, TikTok, Flickr, and so on are great public sources where you can obtain a wide-scale selection of varying images.

With all of that said, the field is advancing daily in its efforts to precisely extract emotion and intent from facial expressions. Companies such as Amazon, Microsoft, and IBM have all designed such systems. Microsoft supports '**perceived emotion detection**' within its Face API, which claims to identify "*anger, contempt, disgust, fear, happiness, neutral, sadness and surprise." Amazon claims that it can identify all seven emotions and "measure how these things change over time, such as constructing a timeline of the emotions of an actor.*"

Technical Components in Emotional Analysis

Here is a small preview of some technical components a developer will need to involve themselves with to gain a better understanding of the complexity of facial and especially emotional analysis.

Expressions

There are many muscles in your face generating hundreds or thousands of facial expressions of emotion and speech – and there are many different dimensions to that, as well. Additionally, facial expressions can be highly nuanced. For example, a simple twitch of the eye or pause when speaking indicates different things.

Temporal

If you stop thinking about it, your facial expressions and emotions have a temporal aspect. Some emotions take place over time; for example, head and eye movements, eye movements, and so on. In some of these cases, various changes in movement may also be analyzed and processed to determine the intent properly.

Context

There are instances where changes in someone's facial expression may have different meanings depending upon the situational context, how they have reacted to that same situation in the past, etc. The exact change made in one environment may indicate something completely different in another.

In-Cabin Sensing

Some of the more recent implementations and efforts towards emotion recognition can be found in automobiles. Affectiva has an Automotive AI product that offers advanced capabilities such as in-cabin sensing and other use cases. Let's highlight a few.

Detection of Drowsiness

This feature analyzes eye closure, yawn, head pose, and other facial features to determine whether or not the driver is asleep or dozing off.

Occupancy

This detects the exact position of individuals in the vehicle. This allows for enforcement and the informing of safety information and additional features.

Distractions

This analyzes head position/rotation, eye gaze, and cell phone usage while driving to determine if the driver is distracted. The vehicle can then slow down automatically.

Child Seat Monitoring

If a child seat is present in the vehicle and a child was placed in the child seat before the vehicle is put into operation, the system can monitor the child seat to ensure that the child is not left behind in the vehicle after the driver exits the vehicle.

Body Key Points

By analyzing various body points, including joints and others, the system can determine body posture to ensure each passenger and the driver are safely seated in the vehicle during operation.

Object Left Behind

This type of implementation can be used to verify objects like cell phones and other items are not left behind in the vehicle after it has stopped when the engine is turned off, and all passengers have exited the vehicle.

Facial Expression Analysis

This is an interesting area of exploration. By detecting various facial units and positions, this information can be compared against a database of registered cognitive states to understand better and determine each occupant's state, reactions, and intent.

For instance, if the system detects that the driver is arguing with the passenger and appears to be distracted or becoming upset, the vehicle can be automatically made to slow down.

Mood and Emotion Detection

Using facial signals and key body points as mentioned above will allow the system to better understand occupant reactions to things that happen within the vehicle. Thereby, being allowed to customize or tailor content recommendations, change cabin settings, and other things.

Cross-Breeding

Finally, a sort of 'cross-breeding' occurs between facial recognition and voice. Speech2Face has purportedly developed a neural network-based model capable of associating facial and vocal characteristics to create approximate images of speakers from a short audio clip.

Of course, there are many others, but these are just a few to highlight how this technology can be used.

Hardware

- Digital Signage
- Bovia New Cameras
- Edge Reference Architectures
- Architecture Implementations With Kubernetes
- Edge Computing Hardware

Software

- Course5i
- Affectiva
- iMotions - The iMotions software allows you to combine facial coding with other biosensors such as eye-tracking, electrodermal activity, and more to analyze human emotions and physiological responses accurately.
- A Comprehensive List of Facial Recognition Solutions
- FaceMe SDK
- Vapor.io is an emerging player who is dedicated to building the cloud-edge infrastructure.
- Project Volutus is a project that is building the network of distributed Edge data centers by placing thousands of mini-data centers at the base of cell towers, which are directly connected to high-speed wireless networks.
- Crown Castle is a partner and investor to Vapor IO, and its assets contain a mix of about 40,000 tower locations and a large metro fiber footprint. Their goal is to run the cloud-edge by using Project Volutus, eventually.
- Edge Computing Software
- OmnisciDB is labeled as the fastest open-source analytics database in the world. It can use both CPU and GPU power to return SQL queries in milliseconds, even using queries that return billions of rows. This database offers three-tier memory management, query vectorization, rapid query compilation, and support for native SQL. Such a solution can be used with big data analytics for performance in fog computing environments.
- FaceME is an advanced solution that coins itself as the world's top AI Facial Recognition Engine. They have several solutions, among them like Facial Recognition SDK.
- BiometricUpdate.com - Facial recognition solutions list various facial recognition solutions that cover many types of usage scenarios.
- Veriff - Biometric Face Match Verification Software

Red Hat is another interesting option that offers various software products to assist in your transition to the Edge. Red Hat® OpenShift is a container-centric, high-performant, enterprise-grade Kubernetes environment that will help in the portability of where your applications run. Red Hat also offers solutions that facilitate virtual machines and HPC workloads, such as the Red Hat OpenStack® Platform.

Storage and data services play a significant role in Edge Computing. Red Hat Ceph Storage is a product that provides self-healing, massively scalable storage for modern workloads that Edge Computing uses, such as storage-as-a-service, data analytics, AI/ML, and backup and restoration services.

EdgeWorx has developed a very capable IoT platform. Their full stack of technology offers a compelling solution:

Image Credit: Forte Group

One of the most impressive demos is using OpenCV. Their latest SDK allows for identifying and recognizing over four hundred and sixty-eight facial landmarks and claims to do so in real-time!

You should also check out the Facial Recognition offering from Microsoft Azure. Their software allows you to introduce facial recognition capabilities into your applications with relative ease.

Use Cases

Facial recognition has many applications and use cases. Here are just a few examples:

- Banking
- Financial
- Insurance
- Public Safety
- Retail
- Healthcare
- Smart Buildings and Cities
- And more.

Future use cases may include examples such as:

- Automotives in the autonomous vehicle industry. Self-driving cars and service delivery vehicles are two examples.
- Order and delivery systems such as the advanced Amazon ordering systems.
- Advanced biometric security systems.

When dealing with Affective Computing, or Emotion AI, the possibilities are endless. Here are a few examples:

Marketing Communications	Tracking Employee Satisfaction	Attention Deficit Analysis
Marketing Research	Patient Care	Autistic Development Support
Content Optimization and Suggestions	Medical Diagnosis	Testing Game Experiences
Customer Service Intelligent Call Routing	Emotional Support Counselling	Evaluating Casino Gambling Experiences
Recommendations During Calls	Fraud Detection	Government Analysis of Population
Continuous Improvement on Reviews	In-Store Shopping Experience Improvement	Integration with IoT based on emotion
Recruitment	Analysis of Driving Performance with Driving	Adaptive/Immersive Game Experience Enhancement
Employee Training	Measurement of Effective Education	

Challenges

There are many challenges in implementing, deploying, and maintaining a facial/emotion recognition system. As you might expect, these can be categorized as 'Hardware' and 'Software,' for the most part.

Hardware

When dealing with storage, several factors can present challenges including:

Storage

Systems like these require the accumulation and storage of large amounts of images. In line with that, we must ensure that the physical storage hardware has fast enough bus speeds to ensure fast data transfer.

Cameras

With the advent of 3D cameras, the options available span the gamut. Obviously, the more you invest, the higher quality of images you will have, and the potential for extracting greater information exists.

Software

Deployment

Since we are at the Edge, deployment is more complicated than a conventional server or cloud-based implementation. There are many types of IoT architectures, and your particulars may vary. In general, we have to have deployments that will be successful, and if not, can be replaced with an appropriate backup plan. We may also have to temporarily disable or take various Edge devices offline during the deployment.

Maintenance

Monitoring your Edge system is very important. Obviously, the more IoT devices you have, the more points of failure you introduce into your equation. Large-scale distribution systems do not get easier by increasing dimensions!

Next Steps & How To Learn More

Recommended Reading

Nist.gov has sponsored a Facial Recognition Vendor Report where hundreds of vendors and algorithms have been evaluated using comparable metrics. This is a paper with over five hundred pages and contains tremendous information.

Cambridge Press has released a book titled '*Biological and Computer Vision*,' which discusses the difference between biological and computer vision. The book details how evolution has equipped us with a very complicated, visual processing system and how studying it has helped inspire better computer-vision algorithms.

Algorithms Are Not Enough, by Herbert Roitblat – While not specific to facial recognition or emotion, it is an essential read in the field of AI development. *The Atlantic* has written an interesting article that states we're not quite there yet regarding facial emotion, but that may not be what you are led to believe.

The Association for Psychological Science has also written an article explaining the difficulties of inferring emotion from facial recognition and patterns. "It is not possible to confidently infer happiness from a smile, anger from a scowl, or sadness from a frown, as much of current technology tries to do when applying what are mistakenly believed to be the scientific facts," the study concludes.

Bryn Farnsworth, Ph.D., on his blog, writes about the 'Facial Action Coding Systems' (FACS), which discusses how facial muscles are used to determine displayed emotion. Within this blog, Bryn also has a fascinating table that shows the 'Main Action Units', which are groups of facial muscles used, such as the inner brow-raising, cheek raiser, lip raiser, and many more. Alongside each one is a video displaying the exact movements for each item.

In the *Cognition and Emotion Journal*, Taylor and Francis have created what is known as the Karolinska Directed Emotional Faces, which is a validation study on emotion and cognition. If you are or are planning on being a thought leader in this space, I strongly encourage you to read this McKinsey report on IoT computing and digital transformations.

Courses & Certifications

- Coursera - What is Facial Recognition
- Coursera Courses
- OpenCV
- Teaching Robots To See

Demonstrations

- Affectiva Market Emotion Analysis
- Oliver API - Emotion Analysis
- Google - OpenCV

Conclusion

Edge Computing, especially when it comes to facial and emotional recognition, is an incredibly lucrative field that may boom in the next five to ten years. As facial recognition becomes more prevalent, moving to an Edge Computing strategy will become imperative. As we have shown, there are concerns with implementation, such as with any other technology. Still, there are many benefits that offset some of those concerns, such as in the area of security.

It is clear that as the digital revolution continues, these next few years will offer incredible inventions for daily usage. Many of these applications will be on IoT devices, whether it is a smart refrigerator, smart shopping center, or other myriads of possibilities.

This, in turn, implies changes such as cost changes/cost management, development approach changes, infrastructure changes, security risks, and more. There is a cost associated with implementing facial and emotional recognition, and with the correct infrastructure implementations. These costs can increase or decrease depending upon many factors. In the end, we must be aware that the landscape has evolved, and will continue to at a rapid pace.

In addition to this, I suspect that formal regulations and laws will also evolve, albeit at a slower pace. This will certainly keep us on our toes. We must be aware of laws and regulations, hardware and software partners, our current development (technical) resources and capabilities, and costs associated with the changes that are upon us.

We have to be sure our applications are now developed so that they will never go down, as opposed to the present approach of accounting for network outages, bandwidth congestion, excessive latency, and more. By selecting the correct architecture, we will reduce latency, congestion, and as a result, eliminate the worry about latency as we move forward.

The future is promising, both in terms of capabilities and costs. All we need to do is ensure we are as prepared for this shift as possible. By identifying all of the factors we discussed, and addressing each in turn, we are in a better position to succeed.

References

1. World's best facial recognition
2. *Forbes.com* - Edge vs cloud edge
3. Edge computing - a beginners guide
4. *Automationworld.com* - Fog computing vs edge computing
5. Github - AWS samples
6. *Affective Computing* - MIT #321, Revised November 26, 1995
7. *Affective* - Automotive AI
8. *MITSloan* - How emotion sensing technology can reshape the workplace
9. *Culturenet* - A deep learning approach for engagement intensity estimation from face images of children with autism
10. Rudovic, O., Lee, J., Mascarell-Maricic, L., Schuller, B., Picard, R. "Measuring Engagement in Robot-assisted Autism Therapy: A Cross-cultural Study". Frontiers in Robotics and AI (2017).
11. Emotional Expressions Reconsidered: Challenges to Inferring Emotion from Human Facial Movements - article. Original article: Barrett, L. F., Adolphs, R., Marsella, S., Martinez, A. M., & Pollak, S. D. (2019). Emotional expressions reconsidered: Challenges to inferring emotion from human facial movements. Psychological Science in the Public Interest, 20, 1–68. doi:10.1177/1529100619832930.
12. Ellen Goeleven, Rudi De Raedt, Lemke Leyman & Bruno Verschuere (2008) The Karolinska Directed Emotional Faces: A validation study, Cognition and Emotion, 22:6, 1094-1118, DOI: 10.1080/02699930701626582.
13. Relationship between emotion models and AI.
14. Kreiman, G. (2021). Biological and Computer Vision. In Biological and Computer Vision (pp. I-Ii). Cambridge: Cambridge University Press.
15. NIST Ongoing Face Recognition Vendor Test (FRVT) 3

About the Author

Matt R. Cole is an experienced senior software architect/developer with over 30 years of experience. Matt developed the Voice Over IP system for NASA, which was used for the Space Shuttle and International Space Station, was an early pioneer of Voice Over IP, developing Microservices within the .Net ecosystem. Matt also has been working with neural networks for over 20 years and focuses a lot of his spare time developing AI/ML applications, especially those targeting neuroscience. Matt specializes in C#, .Net, Azure, and AI/ML.

GlobalLogic
A Hitachi Group Company