Using Principal Component Analysis to Improve Accessibility

Benjamin Simondet
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
simon998@morris.umn.edu

ABSTRACT

In this paper, I describe ways to increase accessibility of computing systems by using technology based on Principal Component Analysis (PCA). While computing systems represent a significant opportunity to increase accessibility, there are currently many ways that physical and mental disabilities may impair a user's experience. The most common PCA-based methods used to reduce the impact of these impairments include facial recognition, emotion recognition, and eye tracking.

Keywords

PCA (Principal Component Analysis), facial recognition, emotion recognition, eye tracking, accessibility

1. INTRODUCTION

When thinking of computing systems in relation to accessibility, the user experience is frequently impacted by a variety of disabilities. Physical disabilities, such as vision loss, cognitive impairments, and limb impairment, can lead to significant problems when using most modern devices. For example, a person with limited motor control may have a lot of difficulty effectively using a touch screen device or other devices with keyboard and mouse input.

In addition to physical disabilities, mental or learning disabilities such as attention deficit disorder or dyslexia can make reading from a screen and navigating through common user interfaces a big challenge. There are some accessibility settings on most major operating systems, but usually these features are modest and only affect font size, color inversion, and text feedback. These features are a good start, but more can certainly be done.

While improving the user experience for people who experience a variety of impairments is important, this technology can also be employed to improve the day-to-day lives of those who use it. However, a lot of the needs of people that have different abilities seem to be passed over. Decreasing access to common computing systems impacts more than just the lives of the people who are unable to use these common computing systems. Fewer people are able to effectively use the devices, and there is less diversity exposure. Diversity can bring more experience and background into the evolution of computing systems, and can be extremely important for continued success. The purpose of Principal Component Analysis (PCA) is to break down complicated situations into smaller manageable computations, and breaking these problems into components seems to be an ideal solution to incorporating diversity. [8]

In my paper, I first give a description of PCA and its current state. I also thoroughly explain the base PCA algorithm so I can describe the augmented versions in terms of their differences. Then, I describe three example applications that use augmented versions of the PCA algorithm to increase accessibility. These applications include facial recognition, emotion recognition, and eye tracking. I end with an evaluation of the current state of PCA in the context of accessibility and the potential future of the technology.

UMM CSci Senior Seminar Conference, April 2016 Morris, MN.

2. BACKGROUND

Since its inception in the early 1900s, PCA has had a large impact on many areas, including neuroscience, statistics, and computer science. PCA rose in popularity due to its ability to take in large sets of data and reduce them to their most significant and meaningful parts through dimension reduction.

Improvement of accessibility was not the first, or the most common, field to embrace PCA, but PCA is a very suitable method for it due to the data being complicated in a variety of ways. However, some potential difficulties may arise in improving accessibility that come from the variance that exists from person to person. For example, emotion tracking may help people who fall on the autism spectrum by tracking user emotions as well as interpreting the emotions of others the user interacts with. However, autism severity varies, and emotions may be easier or more difficult to spot from person to person. PCA can simplify these processes with its ability to spot minuscule differences in given data.

2.1 Principal Component Analysis

PCA was developed in the early 1900s as a method for data analysis and used for developing predictor models in the 1930s. Many statistical and computing methods evolved from PCA for other purposes and are still closely related, such as factor analysis, canonical correlation analysis, and singular value decomposition. PCA is especially effective for this variety of tasks and settings due to its ability to explain variance and to find structures in data. [10]

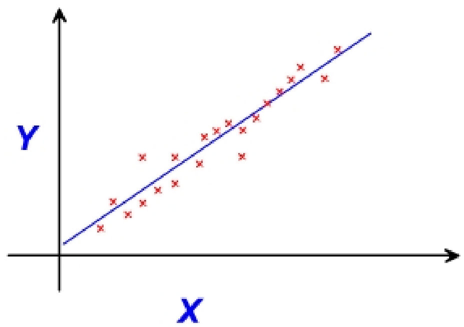

Figure 1: First Principal Component in data [10]

2.1.1 Common Uses

PCA was originally created to be able to simplify a data set. Some common mathematical software packages, such as R Studio, come with a built in PCA algorithm due to the frequency of its use. PCA can be used to remove outliers, reduce the number of dimensions in data, and to clarify any relationships that variables may have. To perform this simplification, principal components in the data are discovered and represented by models. These models represent the amount of variance per dimension of the data. If the data has small eigenvalues, or distance from the first few principal components, some dimensions may be able to be removed. In most data sets, this means the data can be reduced to two to three dimensions, making the data significantly easier to visualize and analyze.

One common use of PCA is facial recognition, also called "Eigenface Recognition". Eigenfaces are the result of doing PCA on a set of human face pictures, meaning that they are essentially the result of combining the similarities of several faces into one. By producing several Eigenfaces, one can construct virtually any human face from various combinations of Eigenfaces. If only one Eigenface is used, differences that an individual person may have from an Eigenface can be used to determine if the same face is processed later, making it particularly useful for facial recognition.

2.1.2 Algorithm

PCA simplifies large sets of data by finding principal components. For a given set of data, there are as many principal components as there are dimensions in the data. For example, a data set showing heights and weights of a group of people has two dimensions. If we also plot age, we have three dimensions in the data. To create the first principal component, we would first find a line in the data to represent the most variance in the data from the mean data point. This will usually be the line that best fits the data. The second would be perpendicular to the first, and should represent the second most amount of variance. In figure 1, the line through the data represents the first principal component because it represents the most variance in the data. The second principal component would be perpendicular to the first, and would represent the second most amount of variance. If age were also plotted, there would be a third line that would represent the third principal component and the third largest amount of the variance.

In any data set, all data points can be recreated with just the principal components. If all eigenvalues, or measures of influence that a data point has on the principal component, are small for an entire principal component, the dimension that was represented by that principal component can be ignored and all points can be projected to the next highest principal component. This can be extremely helpful, especially in cases where visual graphs can be difficult or impossible to draw due to the number of dimensions the data has. There are also benefits to simplifying the data when looking for correlations between dimensions and other variables. [10]

Once all dimensions that are not explaining significant parts of the data have been removed, data can be re-plotted with respect to difference from the first two principal components. This process is done by finding the the eigenvectors of each principal component, or evaluating the eigenvalues for all data represented by each principal component. Data points can then be multiplied by their eigenvalue and re-plotted onto a new set of axes representing up to three of the principal components. Data plotted onto this graph will group together according to the influence, or variance, that they had on the principal components. The groupings that are formed with the re-plotting of the data gives a lot of information about the variables, and can assist in identification and recognition of certain properties. This also allows correlation in variables to become more clear and unusual data to become more obvious. [10]

3. FACIAL RECOGNITION

Facial recognition may be one of the most explored applications of principal component analysis. The PCA algorithm was created to break down models that are complex because of the amount of data that can be attained. When considering human faces, there are more features than can be effectively directly included into an algorithm for facial recognition. Principal component analysis becomes a clear solution when you view the human face as a set of distinct features. If you compare a list of human features to a large set of human faces, there are clear parts that can be ignored in the recognition algorithm due to commonalities. At first glance, it may seem that this would be difficult to apply directly to accessibility due to the lack of current implementations, but as home devices continue to be more connected, the possibilities of PCA applications continue to grow.

3.1 Applications

One example of PCA facial recognition's potential applications to increasing accessibility is the experiment based on the paper "Web-Based Online Embedded Door Access Control and Home Security System Based on Face Recognition" [7]. This is a system that could allow a user who has mobility issues to maintain safety within their home and to allow people into their home without needing to go to the door to do so.

This system allows users to be alerted when there are people at their door, tells the user who the people are if the system has seen the visitors before, and allows the user to lock or unlock their door from any web browser. This has potential uses to anyone with mobility issues, and even has the potential to save lives as it could allow emergency services to enter regardless of the user being able to physically let them in.

This could also have additional uses for users who have impaired memory. By storing visitors' faces into a database, the user could assign names or nicknames to each face to give information about these visitors' identities during future visits.

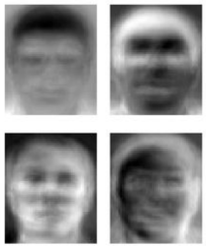

Figure 2: These are a few Eigenface examples. To be effective, they must be combined to form a mean face.[3]

Figure 3: This is an example of a compiled mean face.[3]

3.2 Change in Algorithm

Facial recognition benefits from PCA in many ways, and in most ways the base algorithm is the same. However, there are some image processing steps and algorithm improvements that need to be performed in order for the algorithm to be effective.

For example, distinguishing a face from the surrounding environment is a non-trivial task for PCA. One solution has been presented by Karmakar and Murthy. It introduces a background suppression algorithm and separates out the important features of each face in an automatic cropping algorithm [2].

Karmakar and Murthy accomplish the face isolation by scanning the image pixel by pixel, comparing colors. Pixels determined to be the background are changed to a different color. Then, pixels known to be skin are changed to another color that is distinct from the background color. By determining the size of the known face region, the algorithm can then automatically crop the face section. This leaves a very usable image that can be processed by the aforementioned home security system [2].

A challenge that a small-scale home security PCA algorithm may face comes from constructing the 'mean face' [3]. A mean face is the way the PCA can be applied to facial recognition, and is made by first creating Eigenfaces. Eigenfaces are created by combing a large database of faces to determine similarities that can be disregarded by the algorithm. This is done by finding principal components, or areas with large variance, in faces. Common principal components with low eigenvalues, or little to no deviation from the average, are included to create a set of Eigenfaces. With a complete set of Eigenfaces, any face that has been entered into the algorithm can be recreated [9]. Figure 2 shows an example set of Eigenfaces that may be created. A mean face is created by combining the total set of Eigenfaces, and each face that the system will recognize will be described in terms of its difference from the mean face to allow recognition. Figure 3 shows a compiled mean face. It is clear

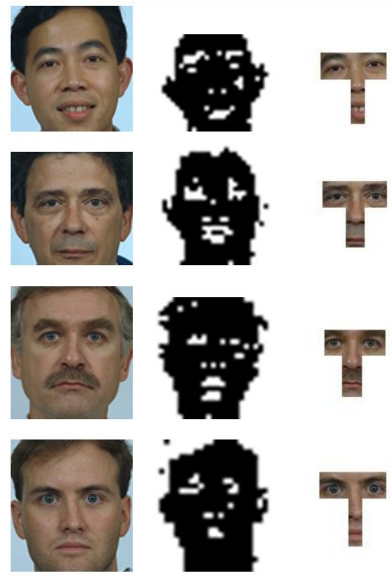

Figure 4: The left column shows a cropped face image. The center column shows the detection algorithm to find key face features. The right column shows the end result of applying the T algorithm. [7]

from the figure that basic facial shape and features appear to be those common to most faces. The recognition algorithm can then disregard these common features of the faces it is asked to identify, and can instead focus on only the unique features. The focus on unique features gives the recognition algorithm a good chance of identifying a face in less time and with more accuracy because there is a smaller amount of data to compare. However, this set of steps can be a challenge to accomplish in a small setting because a user may need to rely on public databases of faces to compile a mean face.

One possible improvement to this method would be to form a 'T' out of each face to reduce the total size of each image being compared [3]. The areas that are most important from the face would be the eyes, mouth, and nose, as these features have the least commonality when comparing to other faces and to the mean face. This region forms a 'T' shape, and reduces the size of the image and amount of processing power required to recognize each face. The 'T' shape can be created in a similar manner as the face cropping described earlier. The background and face are changed to separate colors, like the middle column in figure 4. Then, eyes, mouth, and nose are identified from these simplified pictures, resulting in the 'T' shape in the right column. While there is an overhead to consider that comes with forming the 'T', Karmakar and Murthy found that in practice their 'T' algorithm was computationally more effective than other versions of PCA facial recognition, and was much less complex. [3]

4. EMOTION RECOGNITION

Recognizing emotions in the people that we interact with in everyday life can be considered a crucial skill for interpersonal communication. However, in a study conducted by Rump, Giovannelli, Minshew, and Strauss, both younger and adult-aged participants with autism spectrum disorder (ASD) had significantly more trouble identifying emotion in others in short periods of time [6]. However, they also found that given more time and practice in identifying emotion, individuals with ASD were able to perform as well as individuals without ASD [6]. PCA can be applied to emotion recognition in many different ways due to both its similarity to facial recognition and its ability to be used on a variety of devices in a variety of situations.

4.1 Applications

The most prevalent use of PCA for emotion recognition stems from facial recognition, but includes an additional step to identify emotion. In this case, less emphasis is put on identifying the user's face, but rather common features that may signify that a person is feeling a specific emotion. Emotional facial recognition dictionaries, such as the one produced by Rothkrantz [5], can be a valuable resource in this endeavor. These dictionaries provide a large population of people expressing several different emotions. Similarities between faces representing the same emotions act the same as an individual face in the previous algorithm to identify a specific emotion. Having the same person show several different emotions allows the algorithm to get features that are common from face to face, and to evaluate specific differences from the face that represented showing an emotion.

This can be used by an individual with ASD to identify the emotions of their peers in everyday life through the use of augmented reality devices, as well as practicing to identify pictures of individuals portraying emotions. This practice can allow for significantly increased ability for emotion recognition in individuals with ASD. [6]

An improvement that can be done on evaluation of peer emotions was presented by Yang et al. to evaluate images presented to algorithm within 'one-shot'. [11] This new technique would apply to augmented reality devices that can receive short videos of the user's surroundings. By having a short video of information to process instead of a single picture, the chances that the program will be able to detect the face, evaluate principal components, and compare to eigenvectors of training data increase vastly. This is done by checking frame to frame for the clearest face picture possible. The face can be detected by distinguishing the face from the background using techniques similar to those described in section 3.2. Then, the face that fits the mean face for emotion recognition the most in the short video can be used to determine the subject's emotions.

One particular problem with using PCA for emotion recognition is the large number of faces portraying various emotions that are needed to train the mean faces of the various emotions. This can be solved by using preexisting databases of faces and an algorithm to augment expressions on these faces using methods described by Agarwal, Chatterjee, and Mukherjee [1], which is described in section 4.2.

Figure 5: The dots on these faces show automation points, which are used to modify facial expression. [1]

4.2 Change in Algorithm

In order to adapt face models to specific emotions, a very similar process must be performed on the face models to prepare them as was done in regular facial recognition. The face model must be differentiated from the background by isolating background colors. An additional step must be performed, however, to make a uniform face shape for adaptation. This is important so facial features can be properly changed to reflect emotion in a universal way. To do this, all faces are cropped and re-sized to fit an ovular shape of the same size [1]. To recognize which parts of the face can be cropped, techniques similar to the 'T' algorithm from section 3.2 are used to determine which parts of the face are important and which can be cropped out.

Next, automation points are mapped out on the face models that will be changed. There are 94 total points, with each representing a different muscle group. These points are found in relation to various other facial landmarks, such as the eyes, mouth, and nose [1].

The last main step before facial expression can be modified is training. By having a few face models represent the emotions, the user can specify percentages of each emotion the model is showing. This is important because a face is rarely portraying strictly one emotion. The six emotions that were identified on the training models were joy, surprise, disgust, fear, anger, and sorrow. In figure 5, automation points are shown on two faces while expressing each of several different emotions. By categorizing which emotion the face is representing, the movement of the automation points can symbolize the emotion the person is expressing [1].

Once face models are created with representations of various emotions, the method described in section 3.2 can be used to generate Eigenfaces and a mean face. Then, the user can use the algorithm to determine the emotion of the subject instead of the identity of the subject. This can be used in both augmented reality to identify emotions in real time, and to create labeled pictures that can be used for users to practice identification of emotions on other people.

5. EYE TRACKING

Eye tracking has been studied since the early 1800s to study human thought process. The field has adapted to include studies in psychology, marketing, and product design, and as an input device for computing systems.

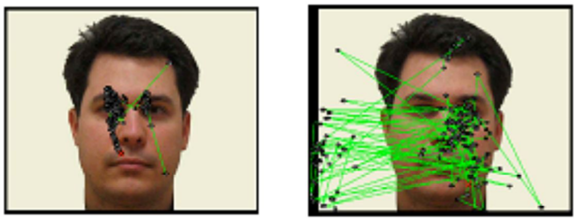

Figure 6: These pictures show two different gaze patterns [4]

Eye tracking can be done in a variety of ways, but the most non-invasive and common involves tracking the center of the pupil. By first asking the user to focus on a certain point on a screen, a camera can be used to accurately figure out where the user is looking. It can also be used to track movement speed and focus placed on an image.

In the case of increasing accessibility, eye tracking can be used in several distinct ways. Eye tracking in ASD detection, as well as detection of other learning disorders such as attention deficit disorder, can be an extremely important part of the diagnosis process and can produce fairly quick results. Eye tracking can also be extremely beneficial to those who are physically impaired by providing an interface that doesn't have any sort of hand or large-muscle requirements. This is done by allowing the cursor to move relative to eye movement, as well as performing left and right button mouse clicks in relation to eye blinks.

5.1 Applications

Eye tracking can be an important method of determining if a person has ASD. This method is so effective because of distinct differences in gaze of individuals with and without ASD. For example, in figure 6 there are two different gaze patterns. The gaze pattern on the left shows a typical gaze pattern. There isn't much variance, and the gaze seems consistent and focused. The gaze pattern on the right shows a common gaze pattern of a person with ASD. The gaze pattern seems to move sporadically, and seems unfocused [4].

There are many different algorithms for evaluating eye tracking, but PCA seems to be a very effective approach in this situation for several reasons. First, there are many different parameters, or dimensions, that can be observed and taken into account with eye tracking. PCA has the ability to both remove some of these dimensions and find correlation in the data, just by using the base PCA algorithm. By plotting the data eigenvalues in relation to the principal components, trends in data can be used to support a diagnosis of ASD.

The ability to track gaze patterns and blinks can also allow a user to interact with a computing system. This method may seem more difficult than typical methods at first, due to the natural movements of eyes. However, after some training, this method can provide an alternative to the usual keyboard and mouse input method. Individuals with a lack of motor control or a limb impairment can strongly benefit from alternative input methods.

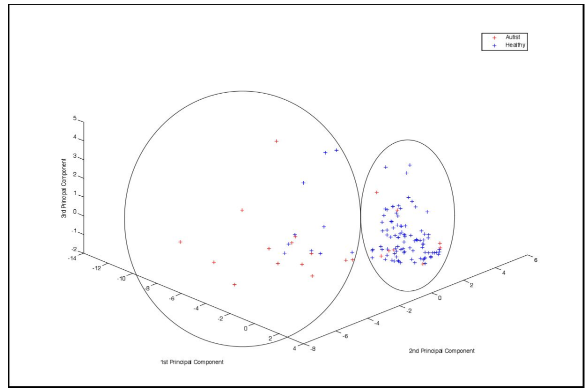

Figure 7: This graph shows the result of PCA on eye tracking data from a small population [4]

5.2 Change in Algorithm and Results

ASD detection from eye tracking can be done with the base statistics PCA algorithm. Because of the many dimensions that can be attained during eye tracking, PCA is a very fitting algorithm to use because PCA helps us find out which dimensions we can ignore. The particular experiment done by Mlouka et al. used 120 different parameters. These parameters measured a wide range of things, such as the time a subject took to focus on an object, or the duration of focus. [4]

After all data was collected, parameters were split into three different groups. These groups were movement data, focus area data, and outside focus area data. After the parameters were sorted, PCA was performed to remove any extra parameters that didn't show any additional data. [4]

PCA revealed that the most important parameters were total gaze distance covered, number of focus points, and gaze spots on and around the regions specified to users. When users were presented pictures of faces, Mlouka et al. found that users without ASD would look at the face's eyes, nose, and then mouth. Users with ASD that were presented with the face pictures looked at random points on the faces and switched the focus of their gaze frequently, as seen in figure 6. [4]

Figure 7 shows the resulting data after the eigenvalues were applied. This is a three dimensional graph that shows the clusters that formed in the data. Data points further to the right represent the least amount gaze movement, and as you move further down and to the left there is more gaze movement and total distance covered. Therefore the users within the left circle would be considered for an ASD diagnosis. This proved to be a more effective measurement in children, which had a 25% error rate, than adults, which had a 37.5% error rate. In this case, error means that the interpretation of a user's data placed them in the cluster that didn't correspond with their diagnosis status. The authors hypothesized that this was due to control over gaze and focus that was gained with practice and experience.[4]

Because there was some error, there is still work to be done with this process and algorithm. For example, out of the 120 different starting parameters, only three are relevant to this interpretation of the data. By researching new parameters and doing further testing, it is possible that the margin of error could be reduced in the diagnosis process.

Additionally, if this study were to be reproduced with more users from a wide variety of backgrounds, there is a chance that there would be a lower margin of error with the preexisting parameters. Regardless of this relatively small margin of error, this algorithm has the potential to allow earlier and more accurate diagnosis of ASD. This can lead to better support and better quality of life from an earlier age.

6. CONCLUSIONS

The methods discussed in this paper show a lot of promise for the overall effectiveness of PCA and its potential for increasing accessibility. The facial recognition example shows the recycling and improvement of excellent past programs that can be used to make a difference in how someone interacts with a computer. Emotion recognition, using a lot of the same technology as facial recognition, can assist those with autism spectrum disorder to identify the emotions of the people around them. Detecting blinks and gestures can allow a person with little to no motor control to use a computer in a way that leverages their limited mobility, and can even assist in the detection of autism spectrum disorder. All of these are significant improvements in accessibility that make a difference for their user base.

However, there is still work that needs to be done. In most of the examples, there are no commercial solutions. Most of these designs were homemade using small-scale equipment. Integration with existing computing systems that are present on a larger scale is important to increase the availability of these solutions to those who need them. As solutions move from cutting edge to more widely available technology, we see an increase in home grown and open source solutions. This is certainly a step in the right direction, but more can still be done.

There are also many future opportunities that PCA may be able to support as home devices continue to improve. For example, smart thermostats that change temperature depending on the amount of movement they detect could be modified to accept gesture control from people who are less mobile. Entertainment systems, namely TVs, could take into account eye tracking and gesture based control to assist those with less motor control or mobility. As technology continues to advance, the possibilities continue to increase.

While possibilities continue to increase, PCA does have some limitations. Because PCA is based on methods that find principal components, reasonably large data sets are needed to find accurate results. Some of these limitations are being overcome through open-source solutions, but there are some cases where this is not possible. For example, facial recognition requires a lot of faces in order to be accurate, and many databases of faces have been made available as a result. In less common situations, such as specific gesture recognition, there may not be any sort of database. This puts the burden of providing this set of data, which needs to match the massive size of some open source face databases in order to be accurate and effective, onto the user or the creator of the system.

Principal Component Analysis has been around for a long time, and has been used in a wide variety of situations. However, new algorithms are beginning to take up the space that PCA has once occupied due to innovations in sensors and 3D image scanning. Fortunately, PCA is finding new light in applications to accessibility that may be extremely useful, due to its open source nature and its ability to be effective without computationally expensive systems, to a field that is growing and is seeking to improve diversity.

Acknowledgments

I would like to thank K.K. Lamberty and Elena Machkasova for all of their guidance and feedback throughout this process. I would also like to thank Paul Schliep and Edison Hach Yellick for their feedback.

7. REFERENCES

[1] S. Agarwal, M. Chatterjee, and D. P. Mukherjee. Synthesis of emotional expressions specific to facial structure. In *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing*, ICVGIP '12, pages 28:1–28:8, New York, NY, USA, 2012. ACM.

[2] D. Karmakar and C. A. Murthy. Face recognition using face-autocropping and facial feature points extraction. In *Proceedings of the 2Nd International Conference on Perception and Machine Intelligence*, PerMIn '15, pages 116–122, New York, NY, USA, 2015. ACM.

[3] S. S. Meher and P. Maben. Face recognition and facial expression identification using pca. In *Advance Computing Conference (IACC), 2014 IEEE International*, pages 1093–1098, Feb 2014.

[4] V. H. G. Mlouka, Martineau. Principal component analysis of eye-tracking data during visual perception of human faces in adults and children with autism. In *International Conference on Advancements of Medicine and Health Care through Technology*, pages 199–204. ResearchGate, 2014.

[5] L. Rothkrantz. Online emotional facial expression dictionary. In *Proceedings of the 15th International Conference on Computer Systems and Technologies*, CompSysTech '14, pages 116–123, New York, NY, USA, 2014. ACM.

[6] M. S. Rump, Giovannelli. The development of emotion recognition in individuals with autism. *PMC*, 1(1):1434–1447, 5 2011.

[7] M. Sahani, C. Nanda, A. K. Sahu, and B. Pattnaik. Web-based online embedded door access control and home security system based on face recognition. In *Circuit, Power and Computing Technologies (ICCPCT), 2015 International Conference on*, pages 1–6, March 2015.

[8] Wikipedia. Computer accessibility — Wikipedia, the free encyclopedia, 2004. [Online; accessed 18-March-2016].

[9] Wikipedia. Eigenface — Wikipedia, the free encyclopedia, 2016. [Online; accessed 18-March-2016].

[10] Wikipedia. Principal component analysis — Wikipedia, the free encyclopedia, 2016. [Online; accessed 18-March-2016].

[11] X. Yang, H. Liu, W. Liu, and Z.-J. Zha. Sparse principle motion component for one-shot gesture recognition. In *Proceedings of the 7th International Conference on Internet Multimedia Computing and Service*, ICIMCS '15, pages 28:1–28:4, New York, NY, USA, 2015. ACM.