Limitations of Local-minima Gaze Prediction

Peter A. C. Varley[a], Stefania Cristina[b], Kenneth P. Camilleri[c] and Alexandra Bonnici[d]
Department of Systems and Control Engineering, University of Malta, Msida MSD 2080, Malta

Keywords: Gaze Prediction, Eye Tracking, Feature Location.

Abstract: We describe a minimal gaze prediction system which is straightforward to implement, can run on everyday hardware, and does not require high-quality video images. We determine head pose and eye gaze from four facial landmarks (nose tip, nose bridge, and eye pupils) which can be expressed as local minima of simple pixel-intensity operations. We assess its stability to: variation of subject's anatomy; facial landmark outliers; and facial landmark small systematic errors.

1 INTRODUCTION

In this paper, we describe a minimal gaze prediction system which is straightforward to implement, can run on everyday hardware, and does not require high-quality video images. We determine head pose and eye gaze from four facial landmarks (nose tip, nose bridge, and eye pupils) which can be expressed as local minima of simple pixel-intensity operations. We assess its stability to: variation of subject's anatomy; facial landmark outliers; and facial landmark small systematic errors.

Our interest is in providing tools for design meetings where designers meet to discuss their ideas, and for client-facing meetings where such designs are displayed to customers. We wish to use gaze to communicate the focus of attention to all participants.

However, many aspects of the current pandemic are unpredictable: not just its duration, but its long-term sociological and cultural effects. We can be fairly sure that indoor meetings will be less frequent, and that when they do take place, the participants will be masked, making gaze-tracking difficult. Outdoor settings are safer and more acceptable, the already-observed shift towards *walking meetings* (Damen et al., 2020) will surely continue, and participants in such meetings may well have their hands full and welcome a hands-free input interface.

We can foresee that social distancing will normalise the idea of seeking information from machines rather than humans, but those seeking information will be wary of touch-screen input interfaces and would prefer something which can be operated from a safe distance, such as the interactive display suggested by (Zhang, 2016).

We can even envisage that, with people spending more time at home, home automation controlled via a wall-mounted screen will become more popular. Controlling this interface by gaze from the other side of the room has its appeal (Tofel, 2020).

Thus, without tying ourselves down to any specific application, we can see various possibilities for a flexible, portable system comprising a projector, a wall-mounted screen, a camera, and a portable computer. The software should be modular—specific applications will have specific requirements, so upgrading any particular software component must be straightforward.

Such a prototype would enable us to determine which potential applications are realistic, and which specific components would require upgrading for these applications to become a reality.

Typical display screens are around 200 × 160cm to 240 × 180cm—in this investigation, we assume 240 × 180cm. A 3 × 3 grid of virtual buttons will be sufficient for simple control applications.

Section 2 describes previous work in gaze recognition in general, and more specifically in locating eye and nose features. Section 3 explains and describes our own system. Section 4 gives snapshots of our results. Section 5 discusses system stability. Section 6 presents our conclusions and recommendations for future work.

[a] https://orcid.org/0000-0003-4181-9234
[b] https://orcid.org/0000-0003-4617-7998
[c] https://orcid.org/0000-0003-0436-6408
[d] https://orcid.org/0000-0002-6580-3424

Varley, P., Cristina, S., Camilleri, K. and Bonnici, A.
Limitations of Local-minima Gaze Prediction.
DOI: 10.5220/0010477400460057
In *Proceedings of the International Conference on Image Processing and Vision Engineering (IMPROVE 2021)*, pages 46-57
ISBN: 978-989-758-511-1
Copyright © 2021 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

2 PREVIOUS WORK

In reviewing previous work, we are interested both in methods and in applications. Choice of a minimal subset of facial landmarks is important, and, as we shall see, methods for locating eyes and noses are of particular interest. Section 2.1 considers applications. Section 2.2 gives an overview of previous work in gaze prediction. Section 2.3 discusses choice of landmarks. Section 2.4 discusses previous work in locating eyes and pupils. Section 2.5 discusses previous work in locating nose landmarks.

2.1 Applications

Toolkits for tracking the gaze of the user of a personal computer are now available commercially—GazeRecorder (GazeRecorder, 2020) is one such—so this must be considered a mature technology. It is nevertheless worth noting that it is strongly range-dependent—GazeRecorder applications are satisfactory when the user is 70–80cm from the camera, but the performance deteriorates rapidly with increasing distance.

Multi-user gaze-tracking applications, and gaze tracking at a distance (anything over 1m), are much rarer.

Zhang (Zhang, 2016) considers various public-facing applications, either outdoors or in shopping malls, based on the concept of an interactive display which is intuitive to use and requires no instruction.

For person-independent eye tracking for public display applications, the accuracy is about one third of the screen size. Rather than attempt a numerical gaze prediction, the system classifies gazes as *left/centre/right*, with N consecutive identical predictions constituting a command to which the system will respond (there is a suggestion that N=6 was used). This is sufficient for an application which allows the user to choose one of three side-by-side options. It was noted that several users wore glasses and used the system without problems, but varying height was a problem as tall users tended to stoop to use the system, while shorter users lifted their heels.

The suggested application is an album cover browser where the user cycles through clockwise or anticlockwise until the desired album shows up (although we note that, even if the intention is to advertise the products of one label, a typical label will have between 1,000 and 10,000 albums on the market, and cycling through all of them is impractical).

A more realistic suggestion is an events calendar for the coming month—users should be able to deduce which way to scroll, and there will not be so many events that they get bored before reaching the one they want.

Sidenmark (Sidenmark et al., 2020) attempt to distinguish natural head movements from intentional head pose changes. Although initial results suggest that this is possible in principle, this remains work in progress.

Mardenbegi (Mardanbegi et al., 2019) addresses a fundamental problem in multi-person gaze-tracking: who is looking at the screen and who is not? Unfortunately, this work includes what to us is a horrible example of a bad interface paradigm: they use shaking the head to signify select. While head gestures is culture-dependent, in most cultures with which we are familiar, nodding signifies acceptance and shaking the head signifies rejection.

2.2 Gaze Tracking (General)

When discussing how of images of faces may be processed, the distinction is often made between *model based* and *appearance based* methods. It is not clear that this distinction is justifiable, let alone helpful, as there is a large overlap.

Model based methods assume that what is being processed is a face, and that faces have certain known properties. Some model based methods hypothesise things which cannot be seen, such as the centre of the eyeball. But methods which only use landmarks which can be seen (eyes, nose, mouth) and label them as component parts of a face are also model based.

Appearance based methods use only that which can be seen. A few (but not many) appearance based methods make no assumptions at all about the face image, but just feed it straight into an AI machine (usually MTCNN). Most appearance based methods compile feature vectors which they then feed into an AI machine (often MTCNN, but Webgazer (Papoutsaki et al., 2016) used SVM).

There are also methods (Ishikawa et al., 2004) (Weidenbacher et al., 2006) (Sapienza and Camilleri, 2014) which are model based in that they label the features they find as eyes, nose, mouth, but also appearance based in that they work entirely with what can be seen.

Instead of dividing ideas into camps, it is more helpful to look at individual methods, see how well they work, and assess their advantages and disadvantages.

For work prior to 2016, we commend OpenFace (Baltrusaitis et al., 2016), an open-source toolkit which implements those gaze tracking ideas current at the time, including MTCNN, HOG/SVM and Haar Cascades.

As a representative example of the current state of the art, we can consider Zhang (Zhang, 2016), which describes a complete gaze-tracking system, from system components, through implementation and integration, to applications, testing and user assessment. It also includes a good general overview of the state of the art at the time.

Zhang's approach was to use a neural network, to which the input was an annotated image accompanied by a selection of features. This raises questions: surely the strength of neutral networks is that they can detect which patterns are important; if we already know what is important in an image, why use a neural network at all?

More recent developments include:

Zhang (Zhang et al., 2019) presents OpenGaze. an open-source toolkit for appearance-based gaze estimation and interaction. OpenGaze is largely a front-end for OpenFace (Baltrusaitis et al., 2016).

Hagihara (Hagihara et al., 2018) creates a mapping between objects in the real world and objects the user looks at. To this end, they present a 3D gaze tracker which tracks depth as well as x-y coordinates. Their implementation requires the user to wear a helmet or eye-tracker.

Mardenbegi (Mardanbegi et al., 2019) use vestibulo-ocular reflex to determine how far away the face is from whatever the user is looking at. But measurements are made using a virtual reality headset, and require accuracies which cannot be achieved using a typical laptop camera.

Although almost all recent systems have been built around neural networks, it appears that diminishing returns are setting in, with each new development being based on a more subtle aspect of the human eye, resulting in a smaller incremental improvement on its predecessor.

While black-box methods such as neural networks have their advantages, they are uninformative. In practice, mere success is insufficient—we want something which works for reasons which we understand, in order that, when it doesn't work, we understand why and can fix (or work around) the problem. Furthermore, even accepting that predictions from neural networks will be somewhat more reliable than those from simpler methods (since the neural network takes much more information into account when making its prediction), this does not necessarily mean that a neural network system will be more reliable. Prediction is one component of the overall system, a simpler but faster component can make far more predictions in the same time, and statistical analysis of many predictions could well lead to better results than dependence on one somewhat better prediction. Only experimental results can determine which gives better results.

For an alternative approach, we must go back to (Kazemi and Sullivan, 2014), which implements (Dollár et al., 2010)'s Cascaded Pose Regression and (Cao et al., 2012)'s Ferns. This approach has proved popular with hobbyists, and has been implemented by (Xu et al., 2015) and (Papoutsaki et al., 2016) amongst others. The key ideas here are (a) that incremental improvement can turn a good estimate into a better one and (b) that "anywhere in the image" is a sufficiently good starting point. While we agree wholeheartedly with (a), we cannot agree with (b)—what appears in the background in any image is beyond our control, and we cannot predict how it may disrupt iterative improvement.

2.3 Choice of Landmarks

How many facial landmarks are required?

The Dlib implementation (King, 2009) of (Kazemi and Sullivan, 2014) locates and tracks 68 facial landmarks, but this is surely excessive. As an alternative, Dlib provides an option for detecting just 5 landmarks: four eye points (inside and outside corners of the left and right eyes) and one nose point (the base of the nasal septum).

FastHpe (Sapienza and Camilleri, 2014) locates four facial features: left and right eyes, nose, and mouth. The precise landmarks are not specified—features are used to detect motion by comparing one frame to the next, not to determine head pose.

Clearly, if we are to perform 3D calculations, we require at least four landmarks, which must not lie in the same plane (all five of the Dlib landmarks are coplanar), and which must be in the rigid part of the face (and not on the mouth, which can move independently).

2.4 Eyes

Two methods for obtaining eye regions stand out: Haar Cascades (Viola and Jones, 2004), and Cascaded Pose Regression (Dollár et al., 2010) and its derivatives. If we prefer the former, it is because there are several good and readily-available Haar cascades for eyes, notably Yu's left- and right-eye cascades (OpenCV, 2015). Asteriadis (Asteriadis et al., 2006) has observed that the lower 60% of the regions returned by Yu's cascades are centred on the pupil.

FastHpe (Sapienza and Camilleri, 2014) uses Haar Cascades. Applications based on the 68-point version of Dlib (King, 2009) use Cascaded Pose Regression. Both detect eyes but not pupils.

(Timm and Barth, 2011) use Haar Cascades to determine an initial region of interest, and follow this with a gradient-following method to determine pupil positions: the pupil is the point at which most gradient vectors cross.

Recent ideas which are worthy of investigation include:

Liu (Liu et al., 2017) introduced a geometric reformulation which maintains the relationship between left and right eyes when head pose changes.

Zhang (Zhang, 2016) introduced Pupil-Canthri-Ratio, which could usefully be included in any approach which compiles localised features.

Cheng (Cheng et al., 2020) identifies the user's dominant eye, and uses that rather than the other one (or a combination of the two) for gaze-tracking. Their results are often (but not always) better than those of previous methods developed by the same authors.

2.5 Noses

Compared with eyes, noses have received comparatively little attention. One might think that, as noses are a consistent and readily-identifiable shape, they would be an ideal application for Haar cascades, but the reality is otherwise—the only readily-available Haar nose cascade, that of (Castrillón et al., 2007), is far from reliable, as it is not clear which nose landmark it detects. Nevertheless, FastHpe (Sapienza and Camilleri, 2014) uses this cascade.

The 68-point version of Dlib (King, 2009), implementing the ideas of (Kazemi and Sullivan, 2014), locates nine nose points: four in a line from the bridge to the tip, and a further five in an arc covering the nostrils. It can be noted that this method is less successful in practice for noses than for other facial landmarks: it is slower to converge, and the results are less accurate. It is also hard-coded, so impossible to modify.

The more recent 5-point version of Dlib locates just one nose point, the base of the nasal septum.

It is for these reason that we prefer simple ideas such as that of Varley (Varley et al., 2021). This maximises a pixel-intensity-difference operation $2M - L - N$ between three squares: M is centred on the nose tip, and L and N are below and either side of it—nose tips protrude from the face and catch the light, whereas nostrils are concave and dark, as can be seen in Figure 1. Understanding how it works means that we are aware of its limitations: although this method is quite good at finding nose tips, it is even better at finding ear lobes, so it must be constrained to a region of interest which includes the centre of the face and excludes the ears.

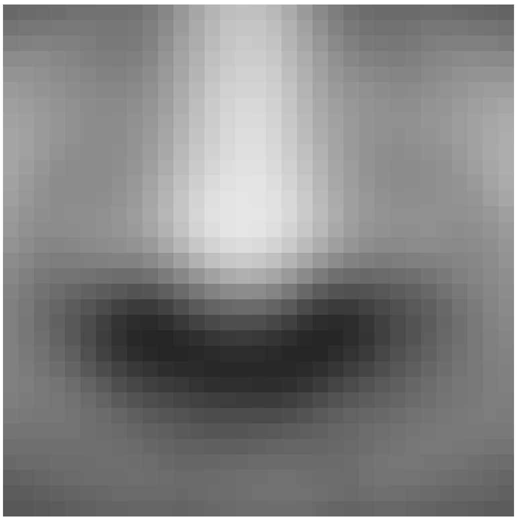

Figure 1: Averaged Nose Tips and Surrounding Regions.

3 IMPLEMENTATION

There are four points on the human face which are mathematically unique: the two pupils, the nose bridge, and the nose tip. The important implication of mathematical uniqueness is that, given a reasonable estimate of where the feature probably is, we can then use optimisation methods to determine a more accurate position.

We make use of these four points as follows:

1. Locate faces in the image. See Section 3.1.
2. Locate the mouth region in each face. See Section 3.2.
3. Locate the eye regions in each face. See Section 3.3.
4. Find the pupils for each eye. See Section 3.4.
5. Find the nose tip. See Section 3.5.
6. Find the nose bridge. See Section 3.6.
7. Calculate the head pose: tilt angle, nod and turn, and eye gaze. See Section 3.7.
8. Predict the gaze target. See Section 3.8.

3.1 Faces

We start our analysis by locating faces in an image. Starting with an RGB image, we take the red channel, which leads to slightly better results than the more usual greyscale. We use Lienhart's Alt2 Frontal Face detector (Lienhart and Maydt, 2002), which in practice we have found to be most reliable, to find the face regions. If several regions are found and they do not

SCIENCE AND TECHNOLOGY PUBLICATIONS

overlap, we process the best of them (regions with two eyes are better than regions with one eye, which are better than regions with no eyes). The face region is used to constrain mouth, eye and nose regions of interest.

3.2 Mouths

We locate a mouth region in each face. As a non-rigid feature, the mouth itself is inappropriate for use in gaze prediction, so these mouth regions are used solely to constrain eye and nose regions of interest, for which mouth regions provide a suitable lower bound.

We use Deniz's Smile detector (OpenCV, 2015) and Castrillón's Mouth detector (Castrillón et al., 2007) to find the mouth region. Although neither of these is entirely reliable, by running both and comparing the results we can usually determine a reliable mouth region.

3.3 Eye Regions

We use Yu's Left Eye and Right Eye detectors (OpenCV, 2015) to find eye regions, and where possible we follow Asteriadis's (Asteriadis et al., 2006) recommendation of using the lower 60% of this region.

Ideally, Yu's cascades will find one left and one right eye. Sometimes they do not, but by applying common sense rules for selecting/estimating missing regions we can usually get a useable result anyway. These rules are:

- If the same detector (left or right eye) detects two eye regions which overlap, merge them
- If a detected left eye overlaps with a detected right eye, remove the one which is on the wrong side of the face
- If there is one left eye and more than one right eye (or vice versa), pick the one nearest the reflection across the face of the unique eye and discard the others
- If there is only one eye, estimate the other one by reflection across the face.

The eye regions are not used directly in calculating gazes, but are one of the best predictors of pupil position, as described next.

3.4 Pupils

The centre of each eye pupil is at the centre of a region with approximate rotational symmetry in which pixel intensity increases with distance from the centre: pupils are darker than irises, and irises are darker than sclerae. (Occasionally, specular reflection may interfere with this general rule.)

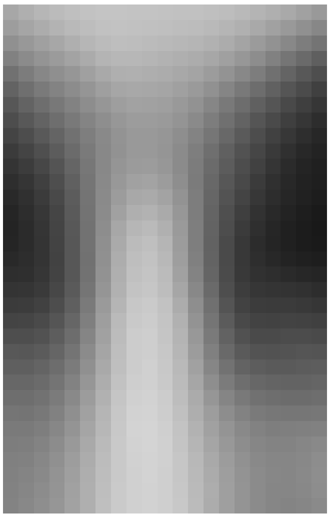

Figure 2: Averaged Nose Bridges and Surrounding Regions.

There are two fairly-reliable methods for locating pupils.

When Yu's cascades find one unambiguous left or right eye (in about 86% of faces), the best estimate of pupil position is the centre of the lower 60% (Asteriadis et al., 2006) of the cascade region.

Alternatively, given a region of interest, the pupil is at the centre of the darkest 5×5 patch in this region. The reliability is around 64%.

By cascading these methods, we can find pupil locations with an accuracy of ±1 pixel and a reliability of around 95%.

3.5 Nose Tips

We use the method described in (Varley et al., 2021). As can be seen in Figure 1, in a typical face, the nose tip is the furthest point on the face from the face plane and catches the light best, and the nostrils are always below and either side of the nose tip and are darker than their surroundings. Even when tightly constrained, the method sometimes finds other nose landmarks rather than tips; when it finds the correct landmark (in about 86% of cases), median accuracy is ±1 pixel.

3.6 Nose Bridges

By virtue of its central position, the nose bridge is the reference point from which all gaze predictions start, as well as contributing to the calculation.

The nose bridge is at the centre of a saddle point, with eye regions to either side, and skin above and below. As can be seen in Figure 2, eye regions are typically occluded by foreheads, noses and cheeks, and are thus dark, while the skin of the forehead and nose is in front of that of the bridge and thus bright.

To locate nose bridges, we use a method conceptually similar to the nose tip method above. We maximise the image intensity differences of four rectangles, $(V+W)-(M+N)$, where V and W are above and below, and M and N are either side of the nose bridge.

As there are potentially several saddle points in a face image, this method must also be constrained to an appropriate region of interest—in practice, an initial estimate that it is somewhere between the eyes is good enough. The method reliably finds the correct landmark, but median accuracy is only ±2 pixels.

3.7 Gazes

We require a system of equations for determining gaze predictions. How many measurable values are there, and how many unknowns?

By locating the landmarks described above, we obtain eight measurable values: the x- and y-coordinates of the four local-minima landmarks. L and R are the left and right pupil coordinates, and V and H are the nose tip and bridge coordinates.

The head is located somewhere in xyz space, where x and y are horizontal and vertical coordinates in the image, and z is distance from the camera.

We model the three human head movements (nodding, shaking, tilting) as (pitch, yaw, roll) of a "disembodied head" (Murphy-Chutorian and Trivedi, 2009) which rotates about a centre point between the eyes and behind the nose bridge. This centre of rotation is located at (X,Y,Z) in xyz space. X and Y have to be determined, but in this analysis we assume that Z can be estimated from anatomical parameters such as the inter-eye distance.

We model the two human eye movements (glancing aside, glancing upwards) as horizontal and vertical translations (in principle, they are pitch and yaw about the centre of the eyeball, but there is little to be gained by modelling this added complexity).

For the purposes of simple analysis, with respect to this centre of rotation, when the head is facing forward (pitch, yaw and roll all 0):

- the left pupil is at $(+E,0,0)$,
- the right pupil is at $(-E,0,0)$,
- the nose bridge is at $(0,0,-B)$,
- the nose tip starts at $(0,+D,-C)$.

This gives us four anatomical parameters:

- E: inter-eye distance,
- B: protrusion of the nose bridge from the face plane,
- C: protrusion of the nose tip from the face plane,
- D: vertical distance from the nose bridge to the nose tip.

Heads have five angular degrees of freedom:

- N is nodding (pitch), which rotates the head in the yz plane, leaving x unchanged,
- S is shaking (yaw), which rotates the head in the xz plane, leaving y unchanged,
- T is tilting (roll), which rotates the head in the xy plane, leaving z unchanged,
- P is glancing aside, which in principle rotates the pupils in the xz plane, leaving y unchanged; we treat it as a translation along the x-axis,
- U is glancing upwards, which in principle rotates the pupils in the yz plane, leaving x unchanged; we treat it as a translation along the y-axis.

Table 1: Notation.

Notation	type	meaning
L	point	left pupil coordinates
R	point	right pupil coordinates
V	point	nose tip coordinates
H	point	nose bridge coordinates
E	length	inter-eye distance
B	length	nose bridge protrusion
C	length	nose tip protrusion
D	length	nose height (tip to bridge)
N	angle	nod (pitch)
S	angle	shake (yaw)
T	angle	tilt (roll)
P	(angle)	glancing aside
U	(angle)	glancing upwards
X	scalar	centre of rotation x
Y	scalar	centre of rotation y

This leaves us with eight equations in eleven unknowns (see Table 1 for a full list of data points and unknowns, and Figure 3 for an illustration). In order to make the problem tractable, we must remove three unknowns, and we choose to do this by estimating other anatomical parameters B,C,D as a fixed proportion of inter-eye distance E. As will be seen in Section 5.1, this can lead to problems when the subjects have particularly small or large noses.

For simplicity, terms of the form $\sin(\alpha)\sin(\beta)$ have been removed as in most cases angles are small.

$$V.x = X - C\sin(S)\cos(N) - D\cos(S)\sin(T) \quad (1)$$

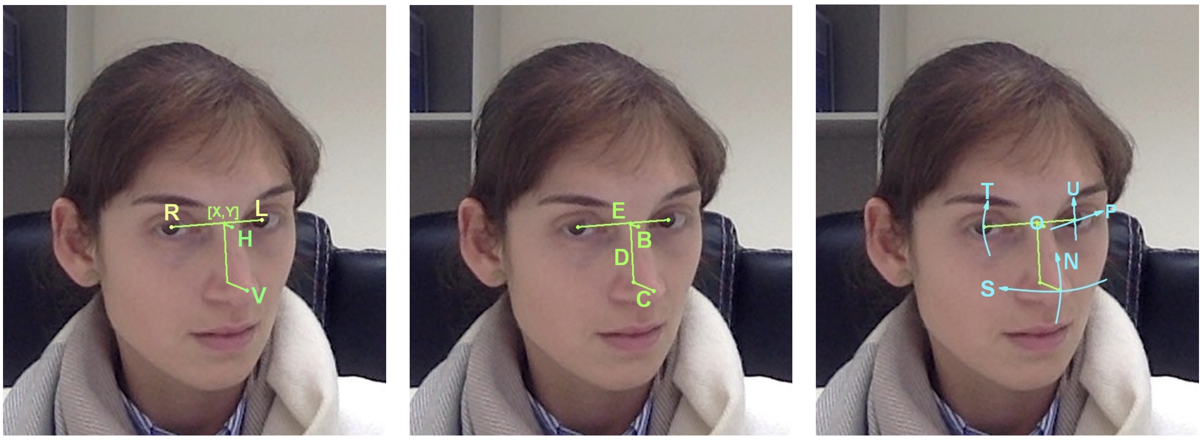

Figure 3: Notation: Points, Distances and Angles.

$$V.y = Y + D\cos(N)\cos(T) - C\sin(N) \tag{2}$$

$$H.x = X - B\sin(S)\cos(N) \tag{3}$$

$$H.y = Y - B\sin(N) \tag{4}$$

$$\begin{aligned} L.x = X &+ (P+E/2)\cos(S)\cos(T) \\ &+ U\cos(S)\sin(T) \end{aligned} \tag{5}$$

$$\begin{aligned} L.y = Y &+ (P+E/2)\cos(N)\sin(T) \\ &- U\cos(N)\cos(T) \end{aligned} \tag{6}$$

$$\begin{aligned} R.x = X &+ (P-E/2)\cos(S)\cos(T) \\ &+ U\cos(S)\sin(T) \end{aligned} \tag{7}$$

$$\begin{aligned} R.y = Y &+ (P-E/2)\cos(N)\sin(T) \\ &- U\cos(N)\cos(T) \end{aligned} \tag{8}$$

Solving this system analytically might be possible, but we prefer a simpler approach. We rearrange the equations into pairs:

$$L.x - R.x = E\cos(S)\cos(T) \tag{9}$$

$$L.y - R.y = E\cos(N)\sin(T) \tag{10}$$

$$\begin{aligned} L.x + R.x = 2X &+ 2P\cos(S)\cos(T) \\ &+ 2U\cos(S)\sin(T) \end{aligned} \tag{11}$$

$$\begin{aligned} L.y + R.y = 2Y &+ 2P\cos(N)\sin(T) \\ &- 2U\cos(N)\cos(T) \end{aligned} \tag{12}$$

$$\begin{aligned} V.x - H.x = &(B-C)\sin(S)\cos(N) \\ &- D\cos(S)\sin(T) \end{aligned} \tag{13}$$

$$V.y - H.y = D\cos(N)\cos(T) - (C-B)\sin(N) \tag{14}$$

$$\begin{aligned} V.x + H.x = &2X - D\cos(S)\sin(T) \\ &- (B+C)\sin(S)\cos(N) \end{aligned} \tag{15}$$

$$\begin{aligned} V.y + H.y = &2Y + D\cos(N)\cos(T) \\ &- (B+C)\sin(N) \end{aligned} \tag{16}$$

This can be solved iteratively (set all cosines to 1 and U to 0 in the first iteration):

$$E = magnitude(L-R)/(\cos(S)\cos(T)) \tag{17}$$

$$\sin(T) = (L.y - R.y)/(E\cos(N)) \tag{18}$$

$$\begin{aligned} \sin(N) = &(V.y - H.y - D\cos(N)\cos(T)) \\ &/(B-C)) \end{aligned} \tag{19}$$

$$\begin{aligned} \sin(S) = &((V.x - H.x) + D\cos(S)\sin(T)) \\ &/((B-C)\cos(N))) \end{aligned} \tag{20}$$

$$\begin{aligned} X = &(V.x + H.x + (B+C)\sin(S)\cos(N) \\ &+ D\cos(S)\sin(T))/2 \end{aligned} \tag{21}$$

$$\begin{aligned} Y = &(V.y + H.y + (B+C)\sin(N) \\ &- D\cos(N)\cos(T))/2 \end{aligned} \tag{22}$$

$$\begin{aligned} P = &((L.x + R.x)/2 - X - U\cos(S)\sin(T)) \\ &/(\cos(S)\cos(T)) \end{aligned} \tag{23}$$

$$\begin{aligned} U = &(Y - (L.y + R.y)/2 + P\cos(N)\sin(T))) \\ &/(\cos(N)\cos(T)) \end{aligned} \tag{24}$$

We have found that, if implemented as-is, this sequence takes some time to converge as it oscillates. However, by smoothing calculation of E (Equation 17) so that $E = (E_1 + E_0)/2$ for previous value E_0 and new value E_1, it converges very quickly. We use 10 iterations, but 4 should be sufficient for stable predictions.

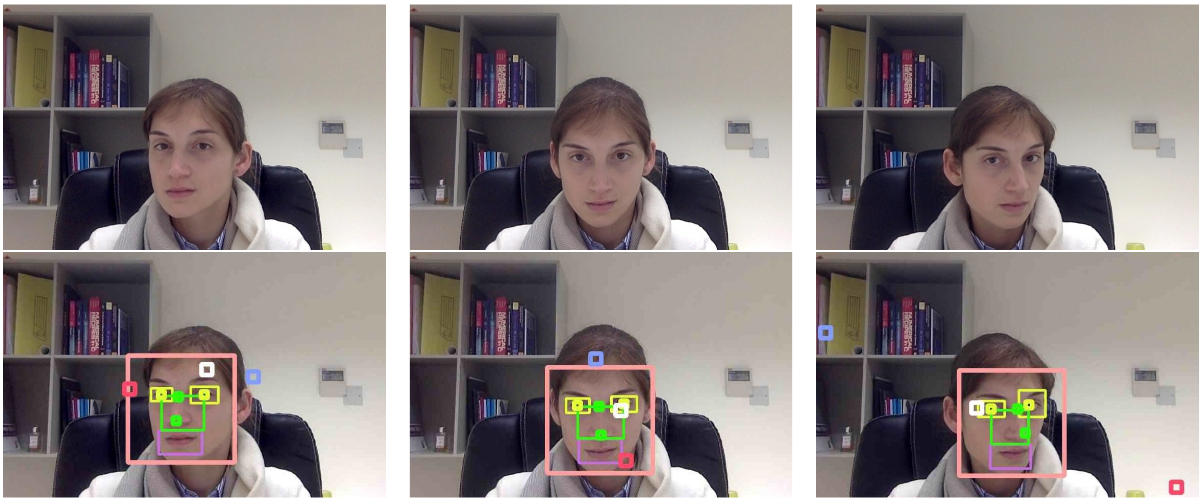

Figure 4: Original and Processed Images: Looking Right, Down, Left.

3.8 Gaze Target

The overall gaze prediction is a vector, the sum of the head pose and eye gaze, originating from the nose bridge.

Geometrically, the head pose (S,N) and pupil displacement (P,U) are 2D projections of 3D vectors originating from (X,Y); we must multiply them by the distance from the subject to the camera to find the gaze target. We estimate this distance from E, the inter-eye distance, as W/E where W is a tuneable program parameter. A further complication is that, anatomically, eye movements are more subtle than head movements, so must be scaled up to obtain the correct effect; scaling factor F is another tuneable parameter. We thus calculate the gaze target G:

$$G = [X,Y] + W([S,N] + F[P,U]) \tag{25}$$

4 RESULTS

The images in the top row of Figure 4 were extracted from a short test video in which the subject's intention was to keep her gaze steady while moving her head, and processed as described in Section 3. The results are as shown in the second row of Figures 4. Images and results have been cropped to remove irrelevant background.

In the results images:

- The pink square shows the face rectangle as found by the face Haar cascade
- The purple rectangle shows the mouth rectangle as determined by the Haar cascades
- The yellow rectangles show the left and right eye rectangles as found by the two eye Haar cascades
- The yellow dots show the predicted positions of the subject's left and right pupils
- The green rectangle shows the region of interest which constrains the search for nose tips
- The green dots show the predicted positions of the subject's nose tip and nose bridge
- The red square marks the predicted head pose vector
- The blue square marks the predicted eye gaze vector
- The white square marks the overall gaze prediction.

For example, in the top left image in Figure 4, the subject has moved her head to the right, and is glancing to her left so as to keep the camera in view. Thus the head pose prediction is to her right; the eye gaze prediction is to her left; and the overall gaze prediction is (relatively) central.

5 STABILITY

We consider three sources of instability which could disrupt our gaze predictions: variations of anatomy; outliers such as those caused by failure to detect landmarks correctly; and small errors such as those imposed by the limitations of pixel resolution.

5.1 Anatomy

We noted in Section 3.7 that estimating anatomical nose parameters as a fixed proportion of inter-eye dis-

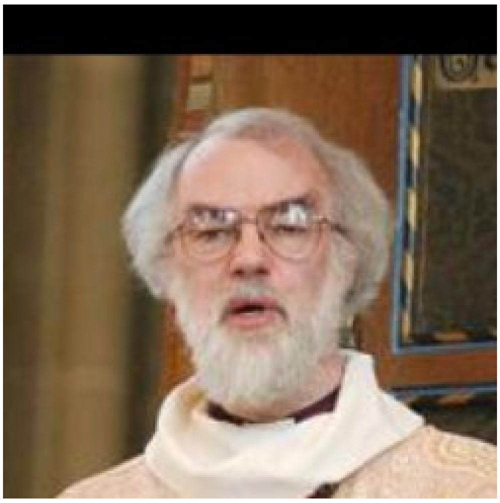

Figure 5: Image: A Short Nose (Huang et al., 2007).

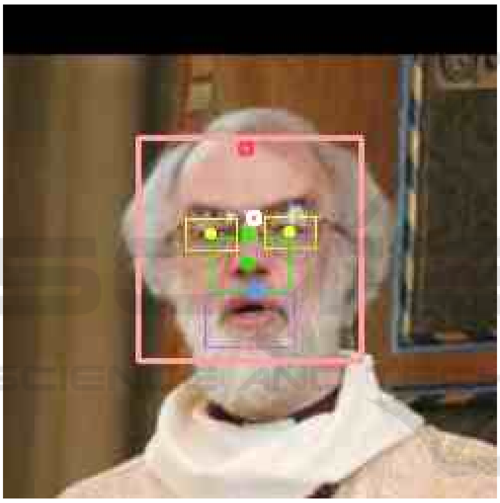

Figure 6: Processed Image: Head Pose Too High.

tance could lead to problems when the subjects have particularly short or long noses. This proves to be the case in practice: short noses appear to be pointing upwards (as in Figures 5 and 6, where $D/E = 0.352$), and long noses appear to be pointing downwards (as in Figures 7 and 8, where $D/E = 0.735$). (Both images are taken from the LFW dataset (Huang et al., 2007).)

This can be overcome by recalibration—the ratios B/E, C/E and D/E are constant for any individual user—but recalibrating for each new user is time-consuming and tedious.

It we want a system which works for everyone straight out of the box, there is no easy solution. In order to allow for the full variety of human noses, we shall need more nose landmarks, and our system and the equations which describe it will inevitably become more complex.

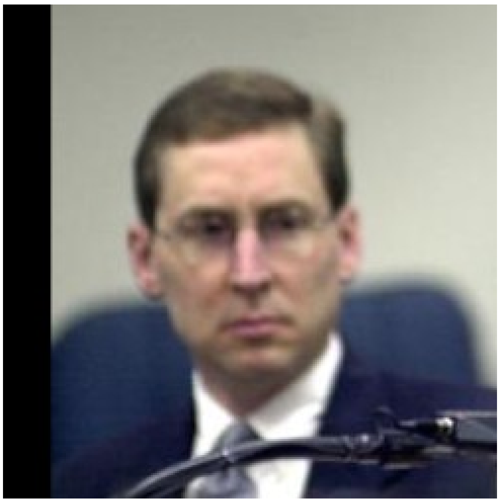

Figure 7: Image: A Long Nose (Huang et al., 2007).

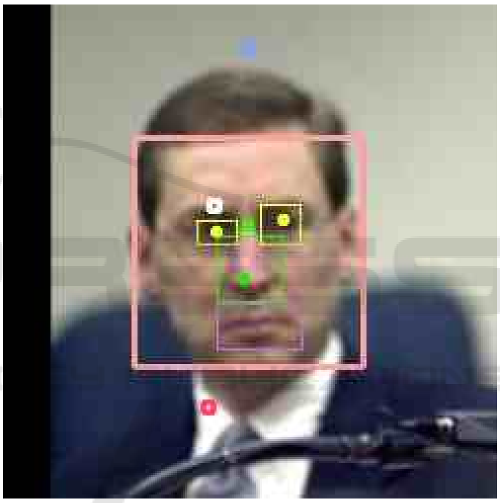

Figure 8: Processed Image: Head Pose Too Low.

5.2 Outliers

Although outliers can occur for any number of reasons, the most common cause in frontal faces is the nose tip finder described in Section 3.5, which in testing found the wrong landmark in about 14% of images. For example, when processing Figure 9, it has found the wing of the nose rather than the nose tip—see Figure 10.

This result is typical: although an outlier in the nose tip prediction has caused a large error in the estimated head pose, the resulting error in eye gaze prediction often almost compensates for this, and the overall resulting error is surprisingly small.

When the head pose is to one side (in our implementation, beyond 28°), the Haar cascades used for finding pupils become unreliable for the more distant eye. Sometimes they fail altogether, and some-

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

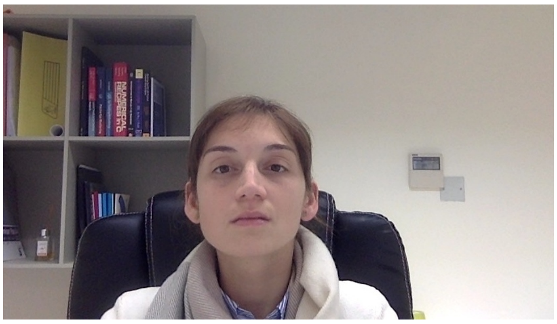

Figure 9: Original Image: Looking Up.

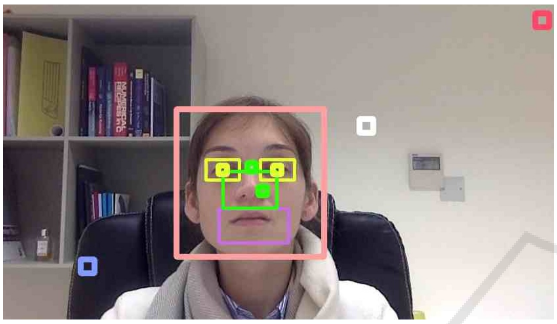

Figure 10: Processed Image: Landmark Failure.

times they return ambiguous regions of interest which can including hair or eyebrows, causing the secondary method for locating pupils to fail too. (Timm and Barth, 2011) have reported similar problems. In such cases, the head pose prediction can be very poor.

5.3 Small Errors

When using a single camera, errors of ±1 pixel are commonplace and, in practice, unavoidable. What effect do they have on gaze prediction?

The subject in Figure 11 (again taken from the LFW dataset (Huang et al., 2007)) has a particularly average nose, $D/E = 0.568$, very close to the median ratio of nose length to inter-eye distance, so is a suitable subject for a sensitivity analysis. We estimate, by comparison with other images, that this image corresponds to a face around 123 cm from the camera.

We find by varying the labelled landmark positions that:

- a 1-pixel x error in the nose bridge position changes the shake angle by 3.23°,
- a 1-pixel y error in the nose bridge position changes the nod angle by 3.35°,
- a 1-pixel x error in the nose tip position changes the shake angle by 3.27°,
- a 1-pixel y error in the nose tip position changes the nod angle by 3.35°.

Figure 11: A Typical Nose (Huang et al., 2007).

These angles will increase with distance, as the size of each landmark in the image decreases. Furthermore, the absolute error on the screen for any given angle error will also increase with distance. How far from the camera can we go before the error becomes unacceptable?

Table 2: Absolute Error Estimates vs Distance.

Distance (cm)	x-error (cm)	y-error (cm)
123	7	7
150	11	10
200	19	19
250	30	29
300	43	42
350	59	57
400	77	75
450	98	95
500	121	118
550	147	143
600	176	171
650	207	202
700	242	236
750	279	272
800	320	311
850	363	354
900	410	399
950	461	448
1000	515	500

Assuming that (a) distance has no other effect on gaze prediction than reducing the size of the face, (b) the angle error resulting from a 1-pixel error increases in proportion to the distance, and (c) the resulting absolute error from a given angle error is in proportion to the distance, we obtain the figures in Table 2.

For example, at 250cm, an error of ±1 pixel can

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

change the nod angle by 6.57° and/or the shake angle by 6.81°, leading to a horizontal error of 30cm and/or a vertical error of 29cm.

Thus, if the target is a 80 × 60 cm box on a 240 × 180 cm screen on the wall of a 10 × 6m room, a 1-pixel error from 300 cm will miss the box, a 1-pixel error from 450–500 cm will miss the screen, and a 1-pixel error from 750–800 cm will miss the wall.

6 CONCLUSIONS

We have shown that a minimal gaze prediction system using only four points can make reasonably reliable predictions for subjects with average noses who sit within 2m of the camera. This system is easily implemented, requiring only four or five Haar cascades (all of which are bundled with OpenCV). It is easy to modify, or even replace, any of the landmark locators.

This simplicity comes at some cost. There are places where we could use more data points, most obviously where we have to make assumptions about the anatomical proportions of the face.

What can be done for people with small or large noses? We could add calibration to retune the system for each new user; the cost is ease of use. Alternatively, the methods for locating nose tips and nose bridges are reasonably reliable, and we could in principle use similar methods to identify other landmarks on the nose, giving us extra equations; the cost is added complexity.

We would also like to be able to weight our calculations so that, when the head is turned, we give priority to the nearer eye. This would be particularly useful in those cases where the head is turned and the location of the more distant eye has not been determined correctly. With only four points, there is no redundancy, and no opportunity to give some points higher weightings than others.

Although it may appear counter-intuitive, gross outliers are not usually a serious problem. In a video-processing system in which landmarks are tracked from one frame to the next, outliers can be caught and discarded.

The most serious problem is that of small errors becoming large errors with increasing distance from the camera, as this imposes a limit on the distance at which gaze prediction can be useful.

On this basis, we can assess the potential applications listed in Section 1. Interactive display boards used from a distance of between 1–2m should certainly be possible. Multi-user interactive boards may be restricted in the number of users, as it will be difficult to place them so that they are less than 2.5m from the board but more than 2m from one another. Sadly, gaze-controlled smart homes may not yet be realistic, as even if the screen is placed on the centre of the longer wall of a 5 × 3m living room, there will be locations in the room which are out of range.

At present, it seems that the best workaround is to improve the hardware: either buy a more expensive camera with higher resolution, or (better still) use multiple cameras.

The natural progression is from still images to video sequences. Before we make this leap, we must ensure that our system is ready for it.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the project: “Setting up of transdisciplinary research and knowledge exchange (TRAKE) complex at the University of Malta (ERDF.01.124)”, which is co-financed by the European Union through the European Regional Development Fund 2014–2020.

REFERENCES

Asteriadis, S., Nikolaidis, N., Hajdu, A., and Pitas, I. (2006). An eye detection algorithm using pixel to edge information. In *ICCVW*.

Baltrusaitis, T., Robinson, P., and Morency, L.-P. (2016). Openface: An open source facial behavior analysis toolkit. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY*, pages 1–10.

Cao, X., Wei, Y., Wen, F., and Sun, J. (2012). Face alignment by explicit shape regression. In *CVPR*, pages 2887–2894.

Castrillón, M., Déniz, O., Hernández, M., and Guerra, C. (2007). Encara2: Real-time detection of multiple faces at different resolutions in video streams. In *Journal of Visual Communication and Image Representation Vol 18 No 2*, pages 130–140.

Cheng, Y., Zhang, X., Lu, F., and Sato, Y. (2020). Gaze estimation by exploring two-eye asymmetry. In *IEEE Transactions on Image Processing (TIP), 29(1)*, pages 5259–5272.

Damen, I., Lallemand, C., Brankaert, R., Brombacher, A., van Wesemae, P., and Vos, S. (2020). Understanding walking meetings: Drivers and barriers. In *ACM Proceedings of CHI 2020*.

Dollár, P., Welinder, P., and Perona, P. (2010). Cascaded pose regression. In *CVPR*, pages 1078–1085.

GazeRecorder (2020). Gazerecorder webcam eye tracking. https://gazerecorder.com/.

Hagihara, K., Taniguchi, K., Abibouraguimane, I., Itoh, Y., Higuchi, K., Otsuka, J., Sugimoto, M., and Sato,

Y. (2018). Object-wise 3d gaze mapping in physical workspace. In *Proc. Augmented Human 2018*, pages 25:1–25:5.

Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst.

Ishikawa, T., Baker, S., Matthews, I., and Kanade, T. (2004). Passive driver gaze tracking with active appearance models. In *Proceedings of the 11th World Congress on Intelligent Transportation Systems*.

Kazemi, V. and Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1867–1874.

King, D. E. (2009). Dlib-ml: A machine learning toolkit. In *Journal of Machine Learning Research 10*, pages 1755–1758.

Lienhart, R. and Maydt, J. (2002). An extended set of haar-like features for rapid object detection. In *Proceedings. 2002 International Conference on Image Processing volume 1*, pages 900–903. IEEE.

Liu, Y., Lee, B.-S., and McKeown, M. (2017). A new reconstruction method in gaze estimation with natural head movement. In *Fifteenth IAPR International Conference on Machine Vision Applications (MVA), May 2017*.

Mardanbegi, D., Clarke, C., and Gellersen, H. (2019). Monocular gaze depth estimation using the vestibulo-ocular reflex. In *Proceedings - ETRA 2019: 2019 ACM Symposium On Eye Tracking Research and Applications*, page 20. ACM.

Murphy-Chutorian, E. and Trivedi, M. M. (2009). Head pose estimation in computer vision: A survey. In *IEEE Transactions on Pattern Analysis and Machine Intelligence Volume 31 No 4*.

OpenCV (2015). *Open Source Computer Vision Library*.

Papoutsaki, A., Sangkloy, P., Laskey, J., Daskalova, N., Huang, J., and Hays, J. (2016). Webgazer: Scalable webcam eye tracking using user interactions. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3839–3845. AAAI.

Sapienza, M. and Camilleri, K. P. (2014). Fasthpe: A recipe for quick head pose estimation. Technical Report TR-SCE-2014-01, University of Malta.

Sidenmark, L., Mardanbegi, D., Ramirez Gomez, A., Clarke, C., and Gellersen, H. (2020). Bimodalgaze: Seamlessly refined pointing with gaze and filtered gestural head movement. In *ETRA '20 Proceedings of the 12th ACM Symposium on Eye Tracking Research and Applications*. ACM, ACM.

Timm, F. and Barth, E. (2011). Accurate eye centre localisation by means of gradients. In *Proceedings. 6th International Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, pages 125–130.

Tofel, K. C. (2020). Eye-gaze tracking on a smart display: The next smart home interface? https://staceyoniot.com/eye-gaze-tracking-on-a-smart-display-the-next-smart-home-interface/.

Varley, P. A., Cristina, S., Bonnici, A., and Camilleri, K. P. (2021). As plain as the nose on your face? In *Proceedings. 16th International Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*.

Viola, P. and Jones, M. (2004). Robust real-time face detection. In *International Journal of Computer Vision, 57(2)*, pages 137–154.

Weidenbacher, U., Layher, G., Bayerl, P., and Neumann, H. (2006). Detection of head pose and gaze direction for human-computer interaction. In *Perception and Interactive Technologies. PIT*.

Xu, P., Ehinger, K. A., Zhang, Y., Finkelstein, A., Kulkarni, S. R., and Xiao, J. (2015). Turkergaze: Crowdsourcing saliency with webcam based eye tracking. Technical Report 1504.06755, arXiv preprint.

Zhang, X., Sugano, Y., and Bulling, A. (2019). Evaluation of appearance-based methods and implications for gaze-based applications. In *Proc. 37th ACM SIGCHI Conference on Human Factors in Computing Systems (CHI 2019)*.

Zhang, Y. (2016). *Eye tracking and gaze interface design for pervasive displays*. PhD thesis, University of Lancaster.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS