Seventh International Conference on Control, Automation,
Robotics And Vision (ICARCV'02), Dec 2002, Singapore

Analogous View Transfer for Gaze Correction in Video Sequences[1]

Tat-Jen Cham
School of Computer Engineering
Nanyang Technological University
Singapore 639798

Shyam Krishnamoorthy
Media Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139

Michael Jones
MERL Cambridge Systems
Mitsubishi Electric
Cambridge, MA 02139

Abstract

This paper provides a framework for doing facial gaze correction in video sequences. The proposed framework involves stages of face registration, face parameter mapping and face synthesis. We introduce the concept of analogous views, and derive a novel formulation which extends view transfers based on epipolar geometry to cope with non-rigid motion. Additionally, a disparity mapping function is derived which is learned from training data and handles both spatial disparities as well as pixel-value changes. The disparity mapping function generalizes to facial expressions, illumination conditions and individuals not in the training set, as shown by the results obtained.

1 Introduction

The use of video conferencing has seen substantial growth over the last few years, particularly over the internet. However, one major hurdle faced in video conferencing is that while speech is enhanced by visual feedback, current systems unfortunately distort the natural communication of facial expressions and language – user experience suffers as a result. Figure 1 illustrates this problem. Since the camera and the screen cannot be co-located, eye-contact will never be established, facial language does not convey naturally, and users may encounter general awkwardness in the conversation. Furthermore in a multi-party conferencing setup, speakers have to declare the person they are speaking to, since other users will not be able to determine the intended gaze direction of the speaker.

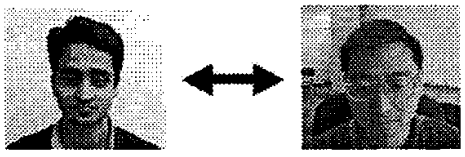


Figure 1: Real-world scenario of video conferencing

This problem is potentially worse if users are video conferencing through their handhelds or other portable devices. The amount of distortion due to poor camera aim, hand jitter and low-quality camera lens will be significantly greater in the captured images. Beyond the novelty factor, it is unlikely that users will find sufficient interest to video conference via portable devices if these problems are not solved.

[1]This work was done when the authors were concurrently working at the Compaq Cambridge Research Laboratory, MA, USA.

There are a number of methods which attempt to address this problem. These are discussed later in section 6.

The contributions of this paper are:

- a novel analysis which extends traditional view transfers based on epipolar geometry to cope with non-rigid body motion;
- a framework which generalizes exemplar-based view transformations to handle different classes of objects with a *single* mapping function;
- a data-driven method which copes with illumination variation and which can be used for reducing the number of input parameters required in view transformation;
- a preliminary system for correcting facial gaze directions of different individuals with different expressions in different illumination conditions.

2 Overview of Gaze Correction Framework

Figure 2 shows a block diagram of our gaze correction framework. The proposed system takes as input a camera image of a person. The output image shows the person for which the face or head region has been modified to have a corrected gaze direction.

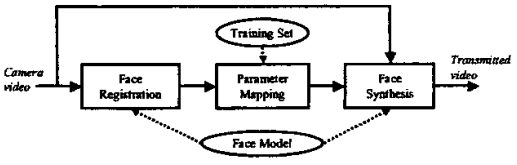


Figure 2: Gaze Correction Block Diagram

The modification is achieved in 3 stages of face registration, parameter mapping and face synthesis which are discussed in more detail below:

Face Registration This stage registers the face in the image to a reference face image. The method used is based on similar work on face models described in [1]. The method establishes pixel-wise correspondences from the input face image to a reference image. These correspondences are expressed in the form of a flow field

$$\begin{aligned} x'_{ij} &= x_{ij} + s^x_{ij} \\ y'_{ij} &= y_{ij} + s^y_{ij} \end{aligned} \quad (1)$$

Authorized licensed use limited to: Antoni Jonjak. Downloaded on June 15,2023 at 16:13:56 UTC from IEEE Xplore. Restrictions apply.

Microsoft Exhibit 1022 - Page 1
Microsoft Corporation v. RealD Spark, LLC
IPR2023-01046 - U.S. Patent No. 10,740,985

Additionally, the model takes into account the gray-level intensities t_{ij} in corresponding pixels. This allows an image to be over-represented by a large vector combining flow field and pixel values.

$$g = [\ \cdots \quad s_{ij}^x \quad s_{ij}^y \quad t_{ij} \quad \cdots\ ] \tag{2}$$

Since the pixel values and flow fields are highly correlated for a face image, g will fall into a low-dimensional subspace, which can be extracted by performing principal component analysis (PCA) on a large amount of training data. Hence the final face image representation can be expressed in a low-dimensional vector f such that

$$g \approx W f \tag{3}$$

where W is a subset of the dominant eigenvectors in the space of g. The advantage of use a face model is that it establishes clean correspondences across the untextured regions of skin on faces. If a normal stereo correlation-based algorithm were to be used, the correspondences will be extremely noisy.

Parameter Mapping The parameter mapping module takes as input the registered face parameters and maps the face parameters to one which corresponds to a gaze-corrected face. The later sections describe the theory behind this, with a detailed description found in section 4.2.

Face Synthesis Here the new face image is synthesized from the gaze-corrected face parameters obtained in the previous stage. The face image is superimposed over the original face.

3 Analogous View Transfer

The system of analogous views is a constrained relation of four views, as shown in figure 3. View V_1 is related to view V_2 in the same way that view V_3 is related to view V_4, although the actual relation ρ_A may not be known. Additionally, V_1 is related to V_3 identically as V_2 to V_4, although again the actual relation ρ_B may be unknown. The system can apply to either views observed in separate cameras, or views observed at different time instances from the same camera, or combinations of both.

The problem of *analogous view transfer* may then be phrased in the following way: *if images are observed in V_1, V_2 and V_3, how can we predict the image that will be seen in V_4?*

One simple example for which this problem may apply concerns video capture with a stereo head/rig. If at a time instance one camera fails to capture an image, is it possible to predict the image that should have been observed, based on the images seen in the previous time instance?

In the case of gaze correction for monocular video conferencing, V_1 and V_3 are views from the actual camera location at different time instances. V_2 and V_4 represent the views from a virtual camera co-located with the video conferencing window on the screen at the same corresponding time instances. For initialization purposes, the user can

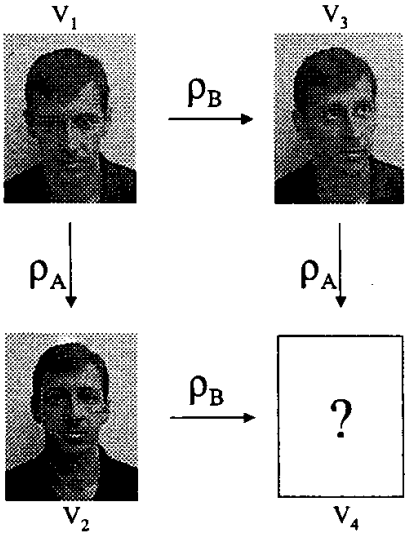


Figure 3: System of Analogous Views

be requested to first look at the conferencing window, for which the image observed in the camera corresponds to V_1. The user can then be requested to look directly at the camera, obtaining the image corresponding to V_2. After the user starts using the system, at any time instance the camera obtains view V_3. The goal is therefore to predict the image in the virtual view V_4.

3.1 Approaches to Analogous View Transfers

The are a number of possible ways to attempt view transfer.

For our gaze correction problem, a naive approach would involve trying to learn a mapping directly between original images of a face to corrected images by using a large number of training images. However, training images do not generalize well because raw images do not form a smooth subspace within image space [1]. The only way that this approach will succeed if all possible pairs of original and corrected face images are provided, which is impossible to achieve.

Alternatively, we can attempt to learn the mapping between original and corrected face parameters computed by the face registration stage. Because the representation involves spatial correspondences as well as pixel intensities, the parameters generalize much better. This method is for example, used by Beymer et al. [2] to generate pose transformations across different faces. However, there are two constraints to this approach: (a) the spatial correspondences must be accurately found between faces of different individuals, not always an easy task especially in different illumination conditions; and (b) the faces in the training set must contain near identical poses and expressions in the test set – particularly difficult if the individuals tested are

Authorized licensed use limited to: Antoni Jonjak. Downloaded on June 15,2023 at 16:13:56 UTC from IEEE Xplore. Restrictions apply.

Microsoft Exhibit 1022 - Page 2
Microsoft Corporation v. RealD Spark, LLC
IPR2023-01046 - U.S. Patent No. 10,740,985

not the same as those in the training set. Additionally, the number of dimensions per face state vector $\boldsymbol{f}$ is still reasonably large (≈ 30). Unless sufficient training images are provided, the learned mapping will be sparse.

In contrast, the approach proposed in this paper does not depend on obtaining spatial correspondences between faces in the test set and the training set. By utilizing epipolar geometry across views of the subject, the method can be directly applied to *dissimilar* objects in the training and test sets. For example, there is no restriction to using our method with a training set of faces, and learning a view-transformation mapping for trees. Furthermore, unlike traditional epipolar view-based transfers, our novel extension incorporates additional views which provide robustness to non-rigid body motion.

3.2 Epipolar Geometry for Rigid Body Motion in Analogous Views

We first consider the case where the relation between views ρ_A and ρ_B are defined by the extrinsic spatial transformation parameters of the corresponding cameras, or rigid body motion of the scene or object observed in the views.

We can specify the transformation from V_1 to V_2 as defined by a rotation matrix $\boldsymbol{R_A}$ and a translation vector $\boldsymbol{T_A}$, and similarly $\boldsymbol{R_B}$ and $\boldsymbol{T_B}$ for V_1 to V_3.

Suppose $\boldsymbol{X_1}$ is a 3D point in the camera coordinate frame of V_1, and $\boldsymbol{X_2}$, $\boldsymbol{X_3}$, $\boldsymbol{X_4}$ represent the same point in the various coordinate frames.

Then between V_1 and V_3

$$\boldsymbol{X_1}^T \boldsymbol{E_B} \boldsymbol{X_3} = 0 \tag{4}$$

where $\boldsymbol{E_B}$ is the well-known product of the translation cross-product matrix and $\boldsymbol{R_B}$. This leads to

$$\begin{aligned} (\boldsymbol{X_2} - \boldsymbol{T_A})^T \boldsymbol{R_A}^T \boldsymbol{E_B} \boldsymbol{R_A} (\boldsymbol{X_4} - \boldsymbol{T_A}) &= 0 \\ \Rightarrow \qquad \boldsymbol{X_2}^T \boldsymbol{E_B} \boldsymbol{X_4} &= 0 \end{aligned} \tag{5}$$

If we consider $\boldsymbol{x_k}$ to be the corresponding image projection of $\boldsymbol{X_k}$ in homogeneous coordinates, we can derive projective epipolar constraints for the two cases:

$$\begin{aligned} \boldsymbol{x_1}^T \boldsymbol{Q_B} \boldsymbol{x_3} &= 0 \\ \boldsymbol{x_2}^T \boldsymbol{Q'_B} \boldsymbol{x_4} &= 0 \end{aligned} \tag{6}$$

where $\boldsymbol{Q_B}$ and $\boldsymbol{Q'_B}$ are the fundamental matrices given by

$$\begin{aligned} \boldsymbol{Q_B} &= \boldsymbol{A_B}^T \boldsymbol{E_B} \boldsymbol{A_B} \\ \boldsymbol{Q'_B} &= \boldsymbol{A'_B}^T \boldsymbol{E_B} \boldsymbol{A'_B} \end{aligned} \tag{7}$$

The differences arise from the intrinsic affine camera matrices for the camera in V_1/V_3 given by $\boldsymbol{A_B}$, and for the camera in V_2/V_4 given by $\boldsymbol{A'_B}$. However, if we assume that apart from focal lengths, the other intrinsic parameters are very similar between the two cameras, then $\boldsymbol{A'_B} = \frac{f'_B}{f_B} \boldsymbol{A_B}$ and $\boldsymbol{Q_B} = \boldsymbol{Q'_B}$ modulo an irrelevant scale factor. We may therefore infer that the epipolar geometry between V_1 and V_3 is almost identical to that between V_2 and V_4.

A similar analysis can be made to show that the epipolar geometry between V_1 and V_2 and that between V_3 and V_4 is identical, i.e.

$$\begin{aligned} \boldsymbol{x_1}^T \boldsymbol{Q_A} \boldsymbol{x_2} &= 0 \\ \boldsymbol{x_3}^T \boldsymbol{Q_A} \boldsymbol{x_4} &= 0 \end{aligned} \tag{8}$$

The above analysis shows that if we know the epipolar geometries relating V_1 to V_2 and V_1 to V_3, we can recover the epipolar geometries of the *parallel view relations* of V_2 to V_4, and V_3 to V_4. This would allow us to apply traditional point transfers by solving for the intersection of epipolar lines. For example, to recover $\boldsymbol{x_4}$ involves solving the linear system given by

$$\begin{bmatrix} \boldsymbol{x_2}^T & \boldsymbol{0}^T \\ \boldsymbol{0}^T & \boldsymbol{x_3}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{Q_B}' \\ \boldsymbol{Q_A} \end{bmatrix} \boldsymbol{x_4} = \boldsymbol{0} \tag{9}$$

Alternatively, the trifocal tensor [3] can also be used.

3.3 View Transfer for Non-Rigid Body Motion

In the case of video conferencing, a substantial amount of the head movement is close to rigid body motion. However, it is not satisfactory to assume as such, especially since the lip and eye movement and the changes of facial expression are important perceptual cues. These are obviously not rigid body motion with respect to the person's head. We can consider non-rigid motion as introducing errors to the epipolar constraint equations.

One point to note is that while these motion are non-rigid over time, the relation of points on an object expressed in different camera coordinates is a rigid body transformation when observed at the same time instance across stereo cameras. This means that (8) is valid regardless of the motion. However, (6) has a non-rigid motion error of $\delta \boldsymbol{x_4}$ such that

$$\begin{aligned} \boldsymbol{x_1}^T \boldsymbol{Q_B} (\boldsymbol{x_3} + \delta \boldsymbol{x_3}) &= 0 \\ \boldsymbol{x_2}^T \boldsymbol{Q'_B} (\boldsymbol{x_4} + \delta \boldsymbol{x_4}) &= 0 \end{aligned} \tag{10}$$

Instead of using only the bottom equations of (8) and (10), we can also combine the top equations and reformulate such that

$$\begin{aligned} (\boldsymbol{x_3} - \boldsymbol{x_1})^T \boldsymbol{Q_A} \{(\boldsymbol{x_4} - \boldsymbol{x_3}) + (\boldsymbol{x_3} - \boldsymbol{x_2})\} &= 0 \\ (\boldsymbol{x_2} - \boldsymbol{x_1})^T \boldsymbol{Q'_B} \{(\boldsymbol{x_4} - \boldsymbol{x_3}) + (\delta \boldsymbol{x_4} - \delta \boldsymbol{x_3})\} &= 0 \end{aligned} \tag{11}$$

In this formulation, the estimation error is a function of $(\delta \boldsymbol{x_4} - \delta \boldsymbol{x_3})$ rather than $\delta \boldsymbol{x_4}$. Depending on the geometrical arrangement of the views, this error may be very small.

In a video conferencing setup, the actual and virtual camera viewpoints typically are not too far apart and have approximate vertical alignment. Hence any non-rigid motion error $\delta \boldsymbol{x_3}$ in V_3 is typically tracked by a similar error $\delta \boldsymbol{x_4}$ in V_4, such that $\delta \boldsymbol{x_4} - \delta \boldsymbol{x_3} \approx 0$.

Why does this work? The essential reason is that four views are used here. ***The additional view provides information redundancy which can be exploited to reduce errors from violation of non-rigidity.***

Authorized licensed use limited to: Antoni Jonjak. Downloaded on June 15,2023 at 16:13:56 UTC from IEEE Xplore. Restrictions apply.

Microsoft Exhibit 1022 - Page 3
Microsoft Corporation v. RealD Spark, LLC
IPR2023-01046 - U.S. Patent No. 10,740,985

Hence we can compute analogous view transfer for non-rigid motion such as facial expression and lip movement from the linear system

$$\begin{bmatrix} (\boldsymbol{x}_2-\boldsymbol{x}_1)^T & \boldsymbol{0}^T \\ \boldsymbol{0}^T & (\boldsymbol{x}_3-\boldsymbol{x}_1)^T \end{bmatrix} \begin{bmatrix} \boldsymbol{Q}_B{}' \\ \boldsymbol{Q}_A \end{bmatrix} \boldsymbol{\Delta}_{43} = \begin{bmatrix} 0 \\ -(\boldsymbol{x}_3-\boldsymbol{x}_1)^T \boldsymbol{Q}_A (\boldsymbol{x}_3-\boldsymbol{x}_2) \end{bmatrix} \quad (12)$$

and $\boldsymbol{x}_4 = \boldsymbol{\Delta}_{43} - \boldsymbol{x}_3$.

Note that the difference between (12) and (9) is a component which *degenerates to zero* under rigid body motion. However under non-rigid body motion, this component helps to *compensate for errors* due to rigid-body assumptions.

4 Data-Driven Estimation versus Analytic Computation for View Transfer

Despite having derived the equations for view transfer, there are two reasons why it is not practical to compute this analytically:

- The epipolar geometry from V_1 to V_3 may change and must be recomputed in full at each time instance, although the actual changes may only involve a few degrees of freedom.
- Gray-level variation in the transferred pixels is ignored. Although the diffuse component of shading should not change from V_3 and V_4, variations will arise due to the specular reflective components of skin as well as shadows.

Instead we propose using a data-driven estimation approach based on training data. However, the previously derived equations are nevertheless useful in distilling the relevant measurements to apply. In particular, (12) can be expressed as a function $\boldsymbol{f}(\cdot)$:

$$(\boldsymbol{x}_4 - \boldsymbol{x}_3) = \boldsymbol{f}(\boldsymbol{x}_3 - \boldsymbol{x}_1,\ \boldsymbol{x}_3 - \boldsymbol{x}_2,\ \boldsymbol{x}_2 - \boldsymbol{x}_1) \quad (13)$$

Note that while $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ are dependent under rigid-body assumptions, all the entries are needed in order to effect computation for non-rigid motion. We also need to account for changes in the epipolar geometry between V_1 and V_3 due to motion. Furthermore, we also wish to model pixel gray-level variations. An augmented function $D(\cdot)$ is therefore proposed, which we call the **Disparity Mapping Function**:

$$\begin{bmatrix} \boldsymbol{x}_4 - \boldsymbol{x}_3 \\ i_4 - i_3 \end{bmatrix} = \boldsymbol{D}(\boldsymbol{p},\ \boldsymbol{v}_3 - \boldsymbol{v}_1,\ \boldsymbol{v}_3 - \boldsymbol{v}_2,\ \boldsymbol{v}_2 - \boldsymbol{v}_1) \quad (14)$$

where

$$\boldsymbol{v}_k = [\ \boldsymbol{x}_k{}^T\ i_k\ ]^T$$

and i_k refers to the gray-level at pixel position $\boldsymbol{x}_k$ in view k, and $\boldsymbol{p}$ is a set of observable/computable parameters which will account for the changes in epipolar geometry between V_1 and V_3 over time.

We call $\boldsymbol{p}$ the **epipolar variable** which should have a maximum of 8 dimensions (corresponding to the 8 parameters in a fundamental matrix) although fewer dimensions are preferred if the motion is expected to be degenerate. Ideally, we should automatically recover the number of dimensions in $\boldsymbol{p}$ (e.g. by performing PCA on training data). However, for simplicity in our case we intuitively decided that an orthographic camera assumption is satisfactory and that the epipolar geometry to vary based on a function of 3D orientation of the face. Hence $\boldsymbol{p}$ has only 3 parameters of eye-to-eye rotation angle, eye-to-eye distance ratio and eye-to-chin distance ratio. The last two ratios are multiplied by the sign of the corresponding disparity in the nose region to disambiguate left/right or top/down gaze directions.

Training instances come in the form of previously observed values of the parameters in (14). The data will be used to model the disparity mapping function as discussed in the next section. Given a single pixel correspondence in V_1, V_2 and V_3 together with their gray-levels, the disparity mapping model will be used to compute the pixel location and gray-level in V_4.

The advantages of using a data-driven disparity mapping function are

- There is no requirement to compute the full fundamental matrices, which is a noise sensitive process. The robustness is provided by limiting the dimensions of $\boldsymbol{p}$.
- There is no requirement to have an analytic relation between fundamental matrices and $\boldsymbol{p}$. This is learned by the disparity mapping model from training data.
- The pixel gray-level variations are also learned directly from the data.

4.1 Semi-Parametric Modeling of Disparity Mapping Function

The training data collected for our gaze correction problem comes in the form of a pair of stereo video sequences. The number of training instances generated for the disparity map modeling is extremely large since each pixel correspondence in each image frame provides a training instance.

Suppose the input parameters are expressed in a vector

$$\boldsymbol{u}^T = [\ \boldsymbol{p}^T \quad (\boldsymbol{v}_3 - \boldsymbol{v}_1)^T \quad (\boldsymbol{v}_3 - \boldsymbol{v}_2)^T \quad (\boldsymbol{v}_2 - \boldsymbol{v}_1)^T\ ]$$

and the output parameters by

$$\boldsymbol{w}^T = [\ \boldsymbol{v}_4 - \boldsymbol{v}_3\ ]$$

Each training instance comprises a pair of input and output parameters.

In order to process the data more efficiently during actual mapping, the the data is first compressed by applying vector quantization to the input part of the data. The vector

Authorized licensed use limited to: Antoni Jonjak. Downloaded on June 15,2023 at 16:13:56 UTC from IEEE Xplore. Restrictions apply.

Microsoft Exhibit 1022 - Page 4
Microsoft Corporation v. RealD Spark, LLC
IPR2023-01046 - U.S. Patent No. 10,740,985

quantization is based on simple clustering to a fixed distance threshold. This generates a set of representatives for the data. In our experiments, we only need to extract 114 representative data points from over 3 million training instances.

Given an initial input probe of $\boldsymbol{u}$, we can estimate the output parameters using *locally-weighted regression* [4] with a planar model. The steps for this is

1. Identify the k representative data points which are closest in input distance to $\boldsymbol{u}$, given by $\{\boldsymbol{u}_j, \boldsymbol{w}_j\}, j = 1 \ldots k$.

2. Compute the linear coefficients $\alpha_1, \ldots, \alpha_k$ for the data points by solving for

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u_1} & \cdots & \boldsymbol{u_k} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{bmatrix} \tag{15}$$

   Depending on the number of representatives k and the dimensions of $\boldsymbol{u}$, the solution for the coefficients may be underconstrained or overconstrained. In either case, the method of singular value decomposition applies equally well.

3. The estimated output parameters may then be computed by

$$\hat{\boldsymbol{w}} = \sum_{j=1}^{k} \alpha_j \boldsymbol{w}_j \tag{16}$$

The use of a planar model for the regression is particularly suitable because for a fixed epipolar variable $\boldsymbol{p}$, the disparity mapping function is inherently linear because of (12), with the only possible non-linearity introduced by the pixel gray-level variation. A locally-linear assumption in this case is expected to be highly satisfactory.

After computing the output parameters for each set of pixel correspondences allows us to synthesize a flow field plus gray-level representation of the corrected image. However, this process is generally noisy and may not form an valid face image. However, we can project these into the face subspace through solving for (3). This generates a valid face image.

4.2 Analogous View Transfer for Gaze Correction

The process to effect analogous view transfer for gaze correction as part of the parameter mapping stage described in section 2 is

1. Initially generate the flow-field and pixel gray-levels of the two reference faces, to get $\boldsymbol{x_1}, \boldsymbol{x_2}, i_1$ and i_2 for each pixel.

2. Loop for each new face parameter $\boldsymbol{f}$:

   (a) Generate the flow-field and pixel gray-levels from $\boldsymbol{f}$ to get $\boldsymbol{x_3}$'s.

   (b) Compute the epipolar variable $\boldsymbol{p}$ from the generated flow-field.

   (c) Loop each set of pixel correspondences between the two reference faces and the input face:

      i. Compute the disparities $\boldsymbol{x_3} - \boldsymbol{x_2}$, $\boldsymbol{x_3} - \boldsymbol{x_1}$, $\boldsymbol{x_2} - \boldsymbol{x_1}$, $i_3 - i_2$, $i_3 - i_1$ and $i_2 - i_1$.

      ii. Compute the new disparities $\boldsymbol{x_4} - \boldsymbol{x_3}$ and $i_4 - i_3$ by using the disparity mapping function $D(\cdot)$.

      iii. The new gaze-corrected flow and gray-level is represented by $\boldsymbol{x_4}$ and i_4.

   (d) Recover a gaze-corrected face parameter $\boldsymbol{f}'$ by projecting the gaze-corrected flow-field and gray-levels back into the face-state subspace. This is to ensure that the generated face is valid.

5 Experimental Results

Training and test data was collected from a single person tracked with an individualized face model, through the use of a set of vertical-baseline stereo cameras. The training data was used to learn the disparity mapping function.

For testing, only data from the top camera is used for the gaze-correction. The results are shown in figure 4. Notice that there is an apparent change in the gaze direction, although there is still significant perceptual differences from the ground truth.



Model-tracked camera view



Synthesized gaze-corrected view



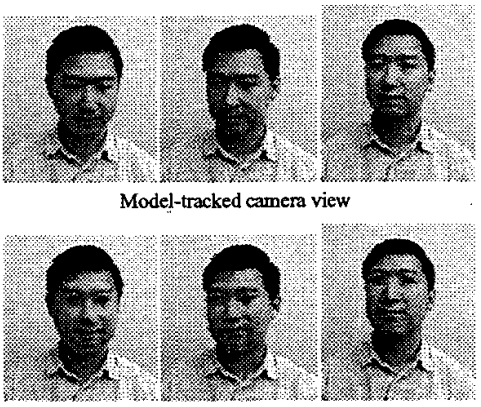
Model-tracked ground truth

Figure 4: Gaze correction results for test images of the first individual.

Authorized licensed use limited to: Antoni Jonjak. Downloaded on June 15,2023 at 16:13:56 UTC from IEEE Xplore. Restrictions apply.

Microsoft Exhibit 1022 - Page 5
Microsoft Corporation v. RealD Spark, LLC
IPR2023-01046 - U.S. Patent No. 10,740,985

Additionally, testing data was collected from a separate individual in a different location with slightly different illumination condition. The results for the gaze correction are shown in figure 5. Once again, there is an obvious change in the gaze direction, although there appears to be additional noise in the estimation.



Model-tracked camera view

Synthesized gaze-corrected view

Figure 5: Results for a separate individual with slightly different illumination conditions. Note that the disparity mapping function used is trained on a different individual. Additionally, no correspondences need to be computed across individuals, only between views of the same person.

Despite the correction in gaze direction, there are still strong perceptual differences from what we would like to see. One problem is simply that the registration only applies to the face. The obvious distortions in the forehead region significantly alters the appearance of the person. Use a full head model (but in 2D) would allow a significant improvement to the realism. Additionally, the results suffer from poor estimation of the iris locations which is a strong perceptual cue to the gaze direction of the person.

6 Gaze Correction Related Work

One approach to partially solving this problem would be to design a system in which a camera and the screen are part of a complex reflection-based setup. Examples of such systems are described in [5]. Such systems are very costly because of the specialized hardware and design required.

Alternatively, there are a number of software-based methods. The method described in [6] makes use of *two* cameras in a stereo setup to create a virtual, geometrically-correct camera view which is coincidental with the screen. Unfortunately, a method which depend on stereo cameras will not have wide applicability since most computers are equipped with at most one camera.

Another approach is described by Zitnick et al. [7]. This involves attempting to align a 3D head model to the camera input images, mapping the image pixels onto the model, rotating the head model to align camera capture and screen view directions, and rendering a virtual view with corrected head pose. However, the 3D model used is either a generic head-shape model or even more simply an ellipsoidal model. The method will not cope well with variation in head shape of different people.

7 Conclusions

This paper addresses the problem of facial gaze correction for video conferencing. A framework is proposed which involves face registration, parameter mapping and synthesis. The concept of analogous view transfer is introduced, and we showed that how the epipolar geometries may be recovered for view transfer. In particular, we provided analysis for both rigid and non-rigid motion, and derived a solution which is robust to non-rigid motion for certain camera configurations. Additionally, we implemented a disparity mapping function which is learned from training data, and which is used to compute the spatial transfer of pixels and changes in pixel gray-levels. This disparity mapping function generalizes to any face shape, illumination condition and across individuals. Results were carried out on two individuals which illustrate this property.

References

[1] M. Jones, "Multidimensional morphable models: A framework for representing and matching object classes," *Int. Journal of Computer Vision*, 1998.

[2] D. Beymer, A. Shashua, and T. Poggio, "Example based image analysis and synthesis," tech. rep., MIT AI Memo 1431, 1993.

[3] S. Avidan and A. Shashua, "Novel view synthesis in tensor space," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, (San Juan, Puerto Rico), pp. 1034–1040, 1997.

[4] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

[5] D. Chen and D. Hartman, "Teleconference system with visual display terminal." US Patent No. 5,953,053, Sept 1999.

[6] J. Lewis, M. Ott, and I. Cox, "Videoconference system using a virtual camera image." US Patent No. 5,359,362, Oct 1994.

[7] C. Zitnick, J. Gemmell, and K. Toyama, "Manipulation of video eye gaze and head orientation for video teleconferencing," Tech. Rep. MSR-TR-99-46, Microsoft Corp, June 1999.

Authorized licensed use limited to: Antoni Jonjak. Downloaded on June 15,2023 at 16:13:56 UTC from IEEE Xplore. Restrictions apply.

Microsoft Exhibit 1022 - Page 6
Microsoft Corporation v. RealD Spark, LLC
IPR2023-01046 - U.S. Patent No. 10,740,985