**\section**{Introduction}

Expression mimicry is a key part of and social interaction (REF 2) and communication strategy learning (REF 1). Unconscious imitation can reinforce feelings of affiliation and empathy between people (REF 4). How we learn to imitate is an open question (REF 5), as it involves expression detection and replication. Mimicry links the domains of visual perception, neural coding and translation of internal representations into motor commands. The function of mimicry is often ascribed to mirror neurons (REF 6), which fire during the performance and perception of an action, but such neurons are best considered as component of the process. There is currently a lack of broader computational models that can solve the correspondence problem [REF] underlying mimicry (REF 7).

It would be very useful to be able to objectively evaluate the quality or accuracy of expression mimicry. This requires the decoupling of expression and identity, enabling person-independent characterisation and measurement of expression. Here we present a series of techniques aimed at the computational implementation of expression imitation, the measurement of mimicry accuracy in human subjects, and the removal of personal identity by blending facial characteristics.

The face space formalism (REF 21) has proved an intuitive and useful guess at the brain?s internal representation of facial characteristics. The na\_ve version (REF 22) applies principal component analysis directly to raw image data in image space, producing a new coordinate frame whose axes correspond to the directions of greatest variation in the original data.

The na\_ve PCA technique can be extended, firstly by including morphological data describing faces? shape and secondly by modifying the representation of each face so that it is relative to a mean. A series of portrait photographs can thus be used to create a coordinate frame, which efficiently expresses the variation present in the input set. For ease of usage we will simply call this frame face space, recognising the more generic use of this term elsewhere in the literature. When the input image set comprises diverse configurations of one person?s face, the resulting PCA space is effectively a controllable model of that face.

This technique supports two useful operations: the creation of an artificial video in which a generated face appears to realistically mimic a real face, and the blending together of several face models into a mixed model which resembles all of the source models.

For applications in the study of the perception of dynamic facial expression it would be very valuable to be able to separate facial motion from facial shape by mapping the motion onto an average face, however generating an average dynamic avatar is non trivial. One cannot simply average up multiple faces performing some action under instruction as the timing of the behaviour may differ radically between people leading to temporal misalignment and temporal blur. We have developed a novel method to circumvent this problem. First we build individual expression spaces for multiple actors. We then project frames from a sequence in the multiple expression spaces and average over the result. The sequence of average is then processed by PCA to provide a photorealistic mean expression space.

Avatar realism is key to engineering realistic qualia of personal interaction (REF 23). Previous approach to the generation of photorealistic avatars usually require extensive software engineering effort (REFS 24, 25) and often suffer from the uncanny valley effect (REF 3).

**\section**{PCA modelling of morphed faces}

This section describes the pre-processing and PCA operations necessary to transform a set of input images into a PCA space in detail.

We begin with a set of n h x w input images Ii showing facial portraits of one person effecting different expressions. One image is chosen as the reference image $I\_r$. Each image $I\_i$ is compared with the reference image using a motion estimation algorithm which produces, for each image, a full vector field (one vector per pixel) showing an estimate of the motion between the reference and target images. This is effectively a dense registration relation between $I\_r$ and each $I\_i$. A vector from a point on $I\_r$ shows the new location of that point on $I\_i$, and we term one such set of vectors is termed a warp field.

Each remaining image will be represented by its difference from the reference image, so $I\_r$ should show a neutral expression, with the eyes open and the mouth slightly open, showing the teeth and a small black area between the teeth (so that the motion estimator can find a warp correlation between the reference image and any dental or buccal features in the remaining images).

The motion detector used is the multichannel gradient model (McGM (REF 8,9)), a bio-inspired algorithm which calculates a basis set of spatio-temporal derivatives by convolving

the image sequence with derivative of Gaussian filters, and then combines

them to form derivatives of the Taylor expansion in space and time. Ratios

of the resulting terms then yield robust estimates of image motion between the reference image and each additional frame. In practice, each pair ($I\_r$ and one $I\_i$) are converted to greyscale and subsampled at several different resolutions before submission to the McGM. The resulting lower-resolution warp fields are combined into one field the same size as the $I\_i$, which gives better results than a single full-scale motion analysis.

The following constraints have been helpful. Rigid head movement (translation or rotation) should be kept to a minimum, either by recording protocol or image registration by face detection (good results have been obtained with FaceAPI (REF)), so that the warp fields represent expression and not head movement. The background should be a uniform colour so that it does not affect the final model. Exposure and white balance should be kept constant during recording.

Once warp fields have been obtained for all images, the vectors at each pixel are averaged to give the mean warp field.

?

Each image Ii is now expressed as two components (a similar dissociation to Blanz and Vetter?s separation of 3D shape and texture (REF 10):

\begin{itemize}

\item Texture $T\_i$: an image showing the textural component of the face $F\_i$. Anatomical points on this texture are aligned with each other and with the mean face, as $T\_i$ has been warped from $I\_i$ to align it with the mean face.

\item Warp $W\_i$: a warp field (full vector field the same size as $T\_i$) showing how $T\_i$ should be warped in order to realign it with the expression shown in $I\_i$.

Operationally, the warp field is represented not by the relative displacement of each pixel (a true vector coding) but as the new position of each pixel (an absolute coding). This was found to produce better results upon applying PCA.

\end{itemize}

To reconstruct the original image $I\_i$ from the two components, the warp $W\_i$ is simply applied to the texture $T\_i$ (each pixel in $T\_i$ is moved to its new location according to $W\_i$ and the result is interpolated). This process is termed morphing (after the special effect (REF)) and gives a very high-fidelity reconstruction of $I\_i$. The pair {$T\_i$,$W\_i$} is thus known as a morph pair.

This representation effects the complete decoupling of textural from configurational information, which is very useful as it allows computational separation of two concepts which seem cognitively separable. This means they can be manipulated separately, as when the warp field is amplified to caricature an expression (REF 11).

Once each image $I\_i$ has been represented as a morph pair, the data set is nearly ready for principal component analysis. Two final preprocessing steps are performed : serialisation and mean relativisation.

As PCA operates on vectors of reals, not more complex data structures, each morph tuple is serialised into a morph vector. This is done by iterating columnwise across the elements of the texture?s colour channel matrices (R, G and then B), followed by the $x$ and $y$ warp field matrices, and pushing them into a vector.

The final vector is of length $w \times h \times (2 + 3)$, as each pixel in a $w \times h$ image is linked with 2 reals coding the $x$ and $y$ components of $W\_i$ and 3 reals coding the RGB colour channels of $T\_i$. A morph vector contains all the information necessary to reconstruct a face image, which is done by deserialising the vector, displacing the texture pixels by the warp field, and interpolating. The space of all possible morph vectors we refer to as morph space, and it is a superset of the space of all possible images (image space) as every image can be exactly represented in morph space by a texture with zero warp field.

We thus obtain n morph vectors $M\_i$. The mean morph $M\_m$ is found and subtracted from each $M\_i$, giving mean-relative morph vectors $R\_i$. This operation models the assumption that identity changes are cognitively encoded in terms of difference from a stored mean, although this property would of course be implemented differently in the neural substrate than in the model?s source code.

The relation

\begin{equation}

M\_i = R\_i + M\_m

\end{equation}

splits each morph vector (equivalent to a face image) into a constant part and a variable part. This is not done in such a way as to maximise the information content of the constant part and minimise that of the variable; this is too precise an assumption to make about the face processing system. Decoupling is done using what is perhaps the most naive method (finding and subtracting the mean) in order to maximise the likelihood that the face processing system operates in a similar manner.

The relative morph vectors $R\_i$ are submitted directly to principal component analysis, which finds a new orthogonal coordinate frame such that, iteratively, each new axis encodes the maximum possible remaining variance in the data. We term the new frame a PCA space or expression space; \footnote{Face and expression spaces are mathematically identical, but expression is constant in the former and identity is constant in the latter.)} dimensionality is normally between 50 and 100, which allows for very accurate image reconstruction while retaining a high degree of compression (for $200 \times 240$ images, morph space dimensionality is $200 \times 240 \times 5 = 240,000$, so a 100-d expression space has a compression factor of 2400).

Faces can now be expressed in terms of their coordinates in expression space, which is to say their loadings on the principal component axes. Passage from expression space to morph space is done by multiplying by a matrix encoding the embedding of the expression space reference frame in relative morph space, which we term the expression space matrix $P$. Reconstruction of an image from loadings $l$ involves

\begin{enumerate}

\item Projection of expression space coordinates into relative morph space: $R = P \times l$, where $R$ is the relative morph space coordinate vector.

\item Addition of the morph mean to generate absolute morph space coordinates $M = R + M\_m$

\item Image reconstruction by applying warp to texture and interpolation.

\end{enumerate}

A full PCA face model thus consists of

\begin{enumerate}

\item The expression space matrix $P$

\item The morph mean $M\_m$

\item The variances of each principal component (used to generate faces in a realistic probability distribution).

\end{enumerate}

(TABLE showing data sizes and operation times on different hardware)

**\section**{Expression projection}

The twin decouplings of the morph space paradigm permit a very useful operation: the projection of an expression from one face model onto another. Consider two PCA models $A$ and $B$ generated from input images of two different people performing approximately the same sequence of expressions (or similar expressions in a different order).

One might imagine that if facial morphologies are similar, the warp fields for each expression will be similar. If lighting conditions and skin tones are similar, the texture components will also be similar. Once mean warp fields and textures have been found and subtracted, similarity of the relative warps and textures depends only on similarity of actual expressions, not underlying facial attributes (which are subtracted by the relativisation process).

The principal components (in other words, the orientation of expression space in image space) will also be similar; varying the first principal component of model $A$ will produce a similar facial deformation to varying that of model $B$. Provided this assumption is true, we can take an image $I\_a$ of person A, calculate its PC-loadings using $A$, and pass these loadings to model $B$ to reconstruct an image of person B exhibiting a similar expression.

This process relies on the two expression spaces being similarly oriented in common image space (they do not need to have close origins, since the correct morph mean for each model is added and subtracted in each case). This cannot be excactly guaranteed, however, since even if the two input image sequences represent the mimicry of exactly the same expression sequence, differing facial characteristics may lead to the same expression change covering different amounts of variance in morph space. In practice, simply transferring the loadings does not always result in effective expression mimicry.

A more robust mimicry method is the following:

\begin{enumerate}

\item Start with an image $I\_a$ of person A, along with a PCA space $A$ for that individual and a PCA space $B$ for person B.

\item Encode $I\_a$ as an absolute morph vector $M\_a$ and then a relative morph vector $R\_a$ with $ (R\_a = M\_a - M\_{ma})$ \footnote{ $M\_ma$ is the morph mean of $A$.}. A is necessary for both operations as, even in absolute morph space, the shape component is expressed as a warp from the mean warp configuration).

\item Multiply $R\_a$ by the PCA matrix for person B ($P\_b$). This gives a set of PC-loadings in person B's expression space $E\_b$. We have

$L\_b = R\_a \times P\_b$.

%TODO: insert that this is the inner product.

%CHange F to E everywhere.

This step in effect calculates the best possible representation of the variable component of $I\_a$ (encoded by $R\_a$) in $F\_b$, the expression space for person B. Even if $F\_b$ is very differently oriented from $F\_a$, $R\_a$ will still be sensibly reconstructable, even though its PC-loadings in $F\_b$ may be very different than in $F\_a$. The requirement that $Fa$ and $F\_b$ be similarly oriented is thus removed.

\item Generate a new relative morph vector Rb by multiplying the calculated loadings $L\_b$ by person B?s PCA matrix:

$R\_b = P\_b \times L\_b$

This represents the reconstruction of $R\_a$ (the variable component of $I\_a$) in $E\_b$.

\item Add the mean expression $M\_{mb}$ of person B. This represents combining the constant component of model B with the variable component transferred from $I\_a$. We are in effect applying the expression of $I\_a$ to the face of person B. We have $M\_b = R\_b + M\_{mb}$.

\item Reconstruct an image $I\_b$ from $M\_b$ by applying the warp component to the texture component and interpolating.

\end{enumerate}

This procedure is problematic when the source face is a very different shape from the target face, as projecting the source relative morph vector into the target PCA space will lead to an unnatural expression on the target (consider applying the warp of a smile exhibited on a wide, short face to a narrow, tall face). This mismatch can be avoided by transforming the warp and texture components of the morph vector before projection.

This is done by manually placing 3 keypoints on the morph mean image (which are generated during PCA modelling) for source and target identities , one at the centre of each eye and one at the centre of the philtrum. These define a triangle termed the faceframe. An affine transformation between source and target faceframes is defined and applied to the relative warp field and texture; this aligns them with the target faceframe, rendering it meaningful to add them to the target?s morph mean. As there are only 3 keypoints, only the eyes and philtrum are perfectly aligned, but the improvement is still substantial.

**\section**{Avatar generation}

The projection process, as it allows expressions to be replicated across different identities, enables another useful technique: the fusing of PCA models of several different people into one PCA model depicting an artificial identity which does not actually exist. Conceptually and aesthetically, the physical characteristics of this new face are a blend of those of its ingredient faces. Mathematically, the new face is generated by morphing together ingredient faces.

The naive mathematical implementation of blending is finding the mean in a representational space. As Galton found (REF 12), this approach does not succeed if we average in image space, where not every point corresponds to an image. Direct averaging produces ghosted images which are textural but not anatomical hybrids. We must therefore use a space in which every point in the subspace spanned by real data (input faces) corresponds to an anatomically plausible face, and morph space possesses this property. Representing static images in morph space and finding the mean implements identity blending (like the classic morph special effect (REF 13)) [FOOTNOTE: The special effect requires manual keypoint placing, which is done here by the McGM.] in static images. The same can be done for a PCA model by statically blending different expressions and then applying the PCA modelling procedure to the resulting images

a

The following is a detailed description of the avatar generation process.

\begin{enumerate}

\item We begin with $k$ identities, each set $S^k$ containing $n^k$ portrait images of a particular person. These are subject to the same constraints on alignment and image characteristics as the single-identity PCA process described on page (ref).

\item A reference image $r^k$ is chosen for each identity, subject to previously described constraints.

\item A PCA model is generated for each identity, as described. Each model brings with it a mean morph vector and an associated morph mean image.

\item Each morph mean image is displayed and 3 keypoints are manually placed, one at the centre of each eye and one at the centre of the philtrum. These define a triangle termed the faceframe fi.

\item Each input image $I^k\_i$ is projected into $k-1$ other identities, producing $k$ copies of the same expression, which form the set $E^k\_i$. During projection, the morph vector is affinely transformed to bring its faceframe into line with that of the target identity.

\item Generated images are sharpened by convolving with an unsharp filter (REF 26).

\item As images in the set $E^k\_i$ have been projected into different faceframes, they are not aligned, and so a second transformation is performed (this time on image data only, as the generated images have no warp component). Each image in each $E^k\_i$ can be transformed onto either \textit{a)} a reference faceframe $f\_r$ (chosen arbitrarily from among the $k$ models) or \textit{b)} the faceframe corresponding to the source model for this expression ($f\_k$).

\item Once images in each $E^k\_i$ have been aligned, they undergo the PCA modelling process, but without the final step of PC analysis. In other words, they are each motion-compared with a reference image (we choose the image in which the original expression has been projected onto the reference identity $I\_r$); warp fields are generated and resourced to point from the mean warp; textures are reverse warped to align them onto the mean warp; fields and textures are assembled into morph vectors; and the mean morph vector is found. As each $E^k\_i$ contains images of the same expression across multiple identities, the mean morph image will also exhibit that expression. Its identity, however, will have been morphed into a blend of the k original identities.

\item blended identity. The standard PCA modelling process is run on this set; as this only happens once, a new reference image can be chosen in order to obtain the best warp fields. Alternatively, the reference image of the reference model can be used. We obtain a PCA model with a wide range of emotional diversity and a common blended identity.

\end{enumerate}

**\section**{Discussion}

We have described a PCA modelling process allowing compression of face images into a small number of expression space coordinates (principal component loadings), projection of expressions from one face model to another, and generation of an identity-blended avatar. These techniques have already been ap

The avatar generation technique outlined is robust across diverse facial expressions and variations in facial morphology. Realism can be reduced, however, when the McGM is not able to accurately describe facial deformation by warp fields in the initial PCA stage (due to unsuitable illumination or non-smooth facial features such as glasses, piercings or facial hair) or where face shapes are different enough to make alignment difficult during the two transformation stages. As affine transforms are only done based on three keypoints, corresponding facial features will not always be brought into alignment. This is only a major problem if identities vary greatly in head size.

We envisage that future work could compensate for this problem by automatically defining more keypoints using commercially available face recognition software such as FaceAPI (REF) and transforming by arbitrary warping instead of affine transformation. There is a tradeoff between alignment and realism; the more warping is done, the further we move from the original input faces. However, morphological correspondence to the input faces is not the algorithm?s goal, as long as anatomical realism is maintained.

We also anticipate scaling image resolution up from $120 \times 100$, and the addition of subexpression spaces which separately model individual features such as the eyes and mouth before merging them together during reconstruction. This could allow constraints such as rigidity to be placed on specific features such as teeth and eyes. It would also allow comparison of the current (holistic) model with a local feature encoding scheme, which might lead to insights about the degree of holism exhibited by the human encoding (as the face space paradigm led to insights about expression representation (REF).

We now discuss possible applications of computational mimicry and identity blending.

Computational mimicry has two main uses: generation of face stimuli which imitate others, and characterisation of real mimicry by subjects.

Given a pair of PCA models (which are trivial to generate provided the constraints mentioned earlier are satisfied, which is easy in a controlled environment), expressions can easily be projected from source to target. This can be done either for static faces or, by separately projecting each individual frame, for video sequences. Degrees of caricaturing can easily be applied. The use of such stimuli can be imagined as part of diverse psychophysical protocols, such as finding the detection threshold for erroneous mimicry or investigating to what extent caricaturing improves recognition. The ability to project expression means that it can be kept constant across identities, making it easier to investigate effects due to morphology alone.

Natural mimicry can also be measured. Consider a protocol according to which which a subject is asked to mimic a portrait video sequence while a portrait recording is made. we can project the mimicry sequence into the PCA space of the stimulus, giving a sequence of PC loading vectors in the same expression space as the stimulus. This can be compared to the loading sequence of the mimicked video, either visually (by viewing the stimulus and its projection side-by-side) or using spatial distance measures. The comparison could be processing using statistical tests or information theoretic measures such as mutual information or Shannon entropy, allowing us to measure the degree to which the distributed face perception system (REF 18) is capable of imitating accurately.

The ability to generate a blended avatar renders it possible to generate realistic stimuli situated on the plausible side of the uncanny valley (REF 3) but free from real-world identity. This avoids problems of privacy and ethics compliance (subjects do not have to approve the presentation of their faces to other subjects, as their identities are obscured through blending) and experimental bias (when stimuli with different identities are employed and a common stimulus which does not specifically evoke any of them is required, an avatar can be generated).

We can also imagine diverse applications of avatar generation outside the remit of research. The process is not constrained to generating a mean avatar; during the blending step, the output could be shifted away from the average in morph space to change the appearence of the avatar PCA model. Numerous different identities could thus be generated for use in interactive software applications, games or media.

An interesting property displayed by the avatar is the increase in perceived attractiveness of more average faces (REF 19). We have noticed that the more source identities an avatar is built from, the more attractive the resulting PCA model. This could be leveraged in applications where positively-connotated faces can lend an advantage, such animated user interface agents or virtual newsreaders (REF 20).