REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 2001	Final Technical Report

4. TITLE AND SUBTITLE Building Standardized 3D Face Data Base for Face Recognition Applications	5. FUNDING NUMBERS N68171-00M-5590
6. AUTHOR(S) Thomas Vetter, Volker Blanz	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Albert-Ludwigs-Universitat Freiburg, Germany	R&D 9845-EN-01
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES United States Army Research, Development and Standardization Group, London, England	10. SPONSORING/MONITORING AGENCY REPORT NUMBER N68171-00-M-5590

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release. Distribution unlimited. ~~Other requests shall be referred to British Defense Staff, British Embassy, Attn: Scientific Information Officer, 3100 Massachusetts Ave., NW, Washington, DC 20008.~~	12b. DISTRIBUTION CODE A

ABSTRACT (Maximum 200 words)

This is the final report to our contract N68171-00-M-5590 supporting our effort for building a three dimensional face data base in collaboration with the University of South Florida. This collaboration will continue in the future supported by contract N68171 -01 -C-9000.

During the past 9 months we concentrated our work on three main issues: First, we developed of methods and software for processing three-dimensional face data and started a collaboration with the University of Florida to obtain and process a large set of high quality laser scans of human heads that might be valuable for future research on automated face recognition. Second, we derived a morphable face model from a selected subset of the face data collected so far using a method we developed earlier. Third, we now also performed a series of first experiments on using a three-dimensional morphable face model for pose and illumination invariant face recognition. The results of these experiments demonstrated that the morphable model is a promising approach for compensating pose variations up to 90°.

14. SUBJECT TERMS US Army, United Kingdom, 3D face models, Morphable model, Face recognition, Face space	15. NUMBER OF PAGES 16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19, SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18
298-102

20020405 023

R&D 8945-EN-01
DTIC

BUILDING A STANDARDIZED 3D FACE DATA BASE FOR FACE RECOGNITION APPLICATIONS

Final Technical Report
by
Thomas Vetter, Volker Blanz

May 2001

United States Army

EUROPEAN RESEARCH OFFICE OF THE U.S. ARMY

London, England

CONTRACT NUMBER .N68171-00-M-5590

Approved for Public Release; distribution unlimited

Albert-Ludwigs-Universität Freiburg
Germany

AQ F02-06-1081

Content:

I. Procedures for collecting and processing of three-dimensional scans of human heads obtained through laser scanning

page 3

II. Building a Morphable 3D Face

page 6

III. A Morphable Model for Face Recognition

page 7

ANNEX

A: Using the 3D Face Space

B: List of the 50 faces used in the Face Space

Processing Three-dimensional Face Data in Collaboration with the University of South Florida

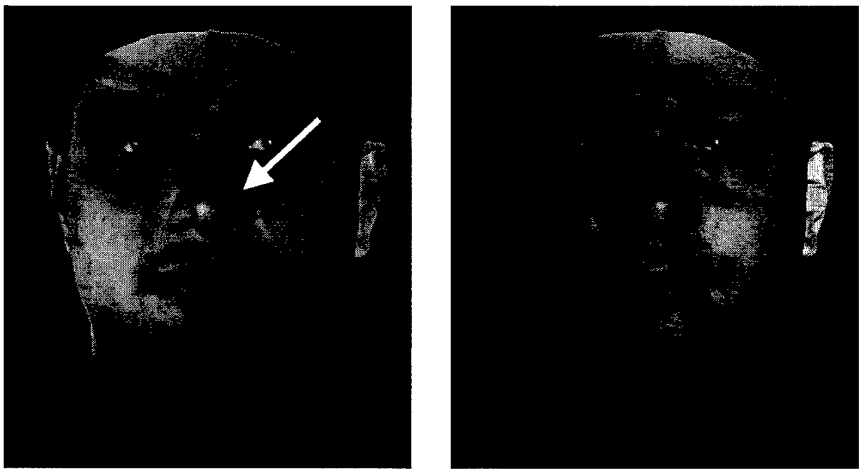
Summary: This is the final report to our contract N68171-00-M-5590 supporting our effort for building a three dimensional face data base in collaboration with the University of South Florida. This collaboration will continue in the future supported by contract N68171-01-C-9000.
During the past 9 months we concentrated our work on three main issues: First, we developed of methods and software for processing three-dimensional face data and started a collaboration with the University of Florida to obtain and process a large set of high quality laser scans of human heads that might be valuable for future research on automated face recognition. Second, we derived a morphable face model from a selected subset of the face data collected so far using a method we developed earlier. Third, we now also performed a series of first experiments on using a three-dimensional morphable face model for pose and illumination invariant face recognition. The results of these experiments demonstrated that the morphable model is a promising approach for compensating pose variations up to 90° .

Keywords: 3D Face models, Morphable Model, Face Recognition, Face Space

I. Procedures for collecting and processing of three-dimensional scans of human heads obtained through laser scanning:

In our first interim report we gave a detailed description of procedures for recording and collecting thee-dimensional imagery of human faces using a Cyberware laser scanner. We also gave a detailed description on methods for post-processing the data to transfer the raw data into a standardized data set of 3D face models of human heads.
Our suggestions for the recording of three-dimensional imagery of faces were only partially applicable by the scanning team at the University of South Florida due to the fact that they used the latest scanner type developed by Cyberware. This new scanner type offers a huge improvement in the spatial resolution of the texture map recorded from the faces and has also a built-in lighting set up for illuminating the faces during scanning. This integrated light set up turned out to be very non flexible and does not offer the possibility to adjust the set-up for our needs. This is especially bad since the light set up as built in results in illumination artefacts on the measured texture maps since the left and the right half of the face are not equally illuminated. The right half of the face contains strong attached shadows. These shadows can hardly be removed synthetically afterwards and therefore lead to physically wrong images when directed light is applied to re-illuminate these face models afterwards (see Fig. 1 right).
The work involved in collecting and processing the data was shared between the University of the South Florida and the University of Freiburg. The main effort, recruiting and scanning people, to build up the standardized data base was taken in Florida. There also the first step in processing the data was done: the spikes, a result of scanning artefacts, were removed from the data. Then the data were sent to

Freiburg to continue and complete the processing. A detailed description of all the individual processing steps was given in the 3rd interim report.
In this report only a short summary is given on the individual processing steps, which consist mainly of an improved stepwise normalization of the data.



Raw texture data under ambient illumination already contain attached shadows.

Illumination with directed light results in physically impossible images.

Fig.1: The non symmetric illumination during recording generates texture maps with clearly visible strong attached shadows (see left images: e.g. on the right side of the nose). When novel illumination conditions such as directed light are simulated, these shadows bound to the texture map lead to images that can not appear under real physical conditions.

A: Aligning the faces in 3D space. In this first processing step the variations in the spatial position and orientation of the individual faces during recording are compensated. The method and motivation for this normalization were described in detail in our 2nd interim report. With this report we also delivered the full procedure *align_3D(...)* performing this normalization in the programming language C. As the result, after this normalization all faces have the same position and orientation in 3D space so the remaining differences between the face are due to the individual differences between the faces and not to global transformations like translation or rotation.
B: Normalizing the face boundaries: The main purpose of the data base is its use for training and testing various face recognition algorithms. To avoid a simple identification through individual scanning artefacts such a hair strand over the ear or a visible part of a particular shirt, all faces are trimmed to an identical face boundary. This guarantees that individual faces can only be recognized through features extracted from the shape and texture of faces and not by the random shape of the boundary of the scan. In figure 2 the cut lines are shown.

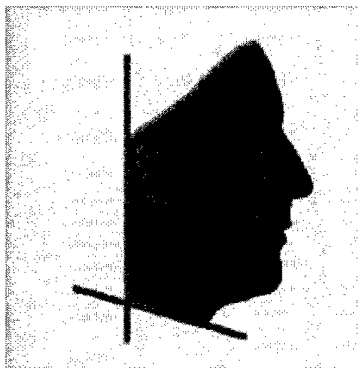


Fig.2: Trimmed boundary of the forehead (green line) and two cutting planes (blue lines)

The alignment of the cutting planes at the neck and the adjustment of the hairline on the forehead was controlled manually by interactive software tools developed at the University of Freiburg. The result (Fig. 3) of the first two normalization procedures is that all faces have a consistent overall appearance, so they can only be

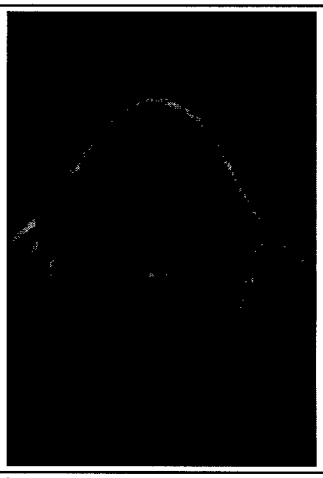
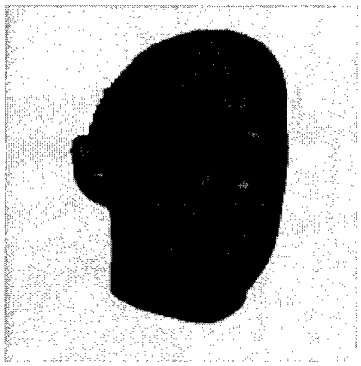
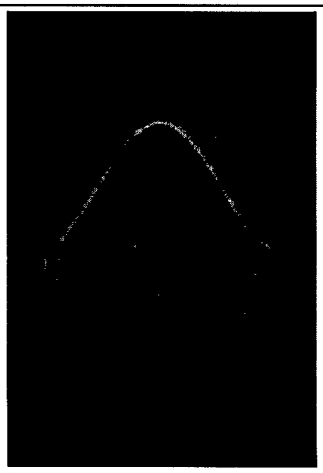
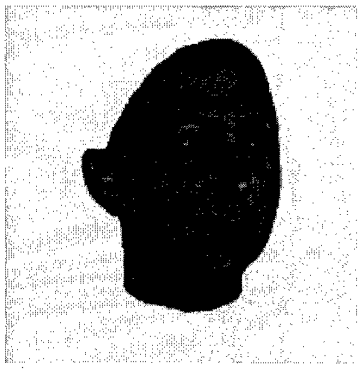
Original Texture	Original Cyberware data	Cut & Aligned Head
		
		

Fig. 3. The transition from the raw scanning data to the face data normalized in the boundary as well as in the spatial orientation and position are show. Examples of the final normalized data are shown in the right column. The left column shows the originally recorded texture map and in the center this texture map is combined with the recorded geometry data and then visualized.

distinguished through internal facial features and not through external features like scanning artefacts, positioning or orientation of the face (see figure 3). However, this normalization of the face boundary could not be performed on all faces that were originally recorded. For example, in some faces the ears were not complete due to the fact that they were covered by hair or by the shower cap which was used to keep hair out of the face region.

II. Building a Morphable 3D Face Model

Setting all faces in correspondence: This final normalization step is different to the others since it can not be applied to single individual faces, but it requires a minimum set of excellent data normalized in steps A & B (see also 3rd interim report). In this last processing step, we normalize the internal data structure of all face scans. While after the first two processing steps as described before (Alignment & Trimming) each face has still its individual number of vertices and texels, we now transform the whole data set into a correspondence-based data structure which fulfils vector space properties. This requires to give up the Cyberware data structure and generate a novel data structure. In this novel structure all faces will have the same number of data points in shape as well as in texture, independent of the actual size of the individual head. That is, in all faces the index numbers within the dataset correspond to corresponding points on the surface, in other words, the tip of the nose of each face has the same index in its vertex list. After this transformation of the data we will be able to apply vector space operations such as adding or subtracting faces and also to compute data correlations on face specific object representation. The software tools for this data normalization we published at the Siggraph´99 conference.

After some delay in the overall data collection and processing pipeline we were able to select a subset of 50 heads of good quality for building up a first morphable 3D face model, to which we also will refer as the *3D Face Space*. All data, documentation and software for a quick visualisation we put on a CD-Rom for distribution (see also Annex A). In a first step, we computed the average of the 50 heads and then the remaining 49 modes of variation derived by a Principal Component Analysis.

We store all this data, the average, the individual principal components as well as the eigen-values in ASCII format to maintain portability between different computer operating systems. From this Faces Space representation, novel faces can be derived by adding the individual principal components to the average. The coefficients to reconstruct the original set of faces which was used to derive the face space is also supplied in ASCII format.

We also supply software in C programming language for performing the combination of the principal components. We supply source code as well as compiled versions for the operating systems Irix, Linux, and Microsoft.

The output of this program (*3dfs.exe*) is a three-dimensional model of a human head. As mentioned earlier, having established correspondence a set of heads can not be represented in the Cyberware data structure which was used earlier to distribute the raw recorded head data. Here we decided to represent the three-dimensional models in the obj data format. This data format was originally developed by Alias Wavefront and has meanwhile become a standard data structure for

representing 3D data for computer graphics applications. It is now also supported by almost all 3D visualization and modelling tools.
For a quick visualization of the head models generated with our software we additionally supply an Obj-viewer based on the platform independent programming language Java. The description of the Obj data format as well as its use in the Obj-viewer source code will enable any interested user to write own applications in all standard programming languages.

III. A Morphable Model for Face Recognition

The face recognition algorithm proposed in this project is based upon the vector space representation of human faces. It describes the three-dimensional shapes and surface textures of human faces in a way such that any vector in the convex hull of a set of examples represents a natural face. In the fully automated technique described in (SIGGRAPH'99), the Morphable Model is learned from a dataset of three-dimensional scans of faces.
Formed by three-dimensional locations and color values of a dense set of surface points, a face vector contains sufficient information to generate synthetic images of the face. Conversely, we presented a method to estimate the face vector from a single image of a person (SIGGRAPH'99), reconstructing three-dimensional shape and texture. The reconstruction algorithm is based on an analysis-by-synthesis loop. In each iteration, the program renders an image of the face, and minimizes the difference of this reconstruction to the original image.
An important property of our approach is that image analysis is not only possible for faces that are in the learning database, but also for novel faces. The reconstruction captures most of the shape and texture of the person in the image, even though not all of the details can be fully recovered.
Reconstruction quality for novel faces depends on the number of free parameters available for optimization. Since each face vector captures information about more than 70000 surface points, it is embedded in a very high dimensional vector space. However, only vectors in the span of the learning faces can be expected to look natural. Hence, the dimension of face space is equal to the number of learning faces. In the face vectors, we separate shape and texture information and form linear combinations separately. Moreover, facial regions such as eyes, nose and mouth are treated separately. Both operations increase the number of free parameters, or vector dimensions, that are available for representing faces. In our experiments, the total number of dimensions of face space is 1000.
We expect that the face vector estimated from an image does not depend on head orientation and illumination conditions. Therefore, it can be used as a representation of individual faces for face recognition.

Recognition Algorithm:

In contrast to the learning database, which consists of three-dimensional face data, we only use image data of the faces that are involved in recognition. The algorithm is given a gallery of front views that are encoded as face vectors and stored for subsequent identification or verification. Novel images, or probe images, that are to be identified or verified, are also transformed to the vector space representation.
The algorithm then compares face vectors according to the following criterion: Based on a Principal Component Analysis, we rescale the face vectors according to the variance within the learning set. Computing Euclidean distance with these rescaled

vectors would give us the Mahalanobis Distance of face vectors. Instead of Mahalanobis Distance, we compute the scalar product between these rescaled vectors after normalizing them to unit length. The result is the cosine of the angle between vectors in face space. Contributions of shape and texture, and of the facial regions such as eyes and mouth, are all weighted equally.
With this measure, identification is achieved by selecting the nearest neighbour, and verification by a comparison with a threshold value.

Fig.4: Test images from the FERET data set used for pose independent face recognition.

Testing Procedure:

The gallery of faces known to the algorithm is a set of frontal views. We then measure performance on recognition from a side view of the person, rotated by 30 to 90 degrees. Thus, none of the gallery views is shown as a probe. We simulate two different tasks:

Identification: Face vectors for all of the gallery images are stored in the program, and given a side view, the task is to decide which of the individuals from the gallery is shown.

Verification: Given a front view from the gallery and a probe image, the task is to decide whether or not both images show the same person.

Synthetic Data: In order to assess performance under precisely controlled conditions, we rendered images of a set of 100 three-dimensional laser scans. None of the scans in this test set was in the learning set of 100 other laser scans that we used to form the vector space. Gallery images are frontal 0° views. Probe images are 30°, 60°, and 90° (profile) views. Images are coloured, and resolution is 256 by 256 pixels. We simulate conditions where the precise orientation, position, size and illumination of heads are unknown by randomizing the starting values of the algorithm that recovers face vectors within a certain range.
FERET Images: The algorithm was run on a subset of the FERET images. We selected 50 front views, and for 23 individuals, we selected an additional side view, with the head rotated by an angle between 30° and 90°. Currently, our algorithm handles only faces with closed mouth, neutral expression, without glasses or hair covering part of the face. However, these restrictions will be addressed in future

work. Starting conditions for the program such as orientation and position of the face, were defined manually, and subsequently refined by the optimization algorithm.

Since none of the individuals from the FERET database was within our set of laser scans, we used all 200 scans for learning the Morphable Face Model. However, after a Principal Component Analysis, we discarded those 50% of the dimensions that contribute least to the overall variance within the data.

Results:

Identification: Identification performance of our algorithm is summarized in Table 1. The percentage in column 6 refers to all trials where the closest individual was in fact the person shown on the probe image. In column 7 and 8, we report the percentage of trials where the correct individual was among the 5% or 10% of the gallery faces that ranked closest to the probe.

Data	Gallery		Probe		Percent		
	Size	View	Size	View	Correct	Within 5%	Within 10%
Synthetic	100	0	100	30	96%	98%	99%
	100	0	100	60	98%	98%	99%
	100	0	100	90	96%	97%	97%
FERET	23	0	23	30-90	91%	91%	100%
	50	0	23	30-90	87%	96%	100%

Table 1: Results for face identification (see text).

Verification:
Figure 5 shows ROC curves for the two sets of images. The stored gallery images were frontal views, and the probe images were side views rotated by 30 degrees to 90 degrees. For the FERET images, the algorithm was tested on those 23 individuals for whom we had computed face vectors for both a front and a side view

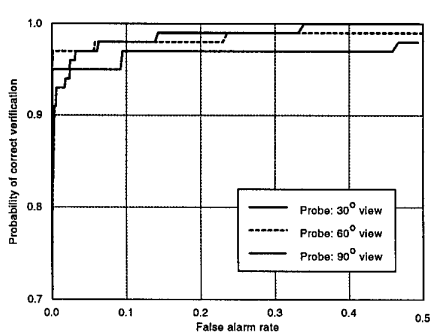


Figure 5: ROC for synthetic data
Probe and gallery size: 100.

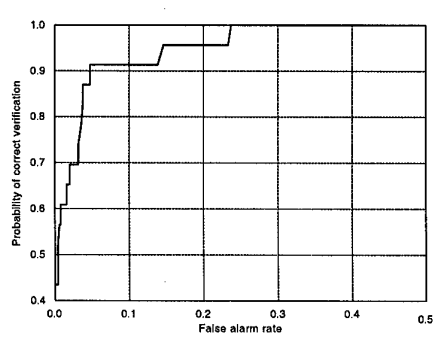


Figure 6: ROC for FERET images
Probe and gallery size: 23

Conclusion:
Based on a relatively small number of images, the results still indicate that our approach is promising. For real images, our method currently requires some manual initialization. However, in many FERET studies, the image coordinates of the eyes are given. Based on this information, much of the manual interaction could be avoided. We also hope to be able to apply our algorithm to a wider variety of images, such as different facial expressions, or partial occlusions. In fact, we believe that it might be exactly these difficult conditions where the power of the Morphable Face Model is manifested.

Reference:

Blanz and T.Vetter, A Morphable Model for the Synthesis of 3D Faces.
In: SIGGRAPH'99 Conference Proceedings, (1999) 187-194.

ANNEX A: Using the 3D Face Space

How to use the 3D Face Space CD

Preliminary: Installing the 3DFS package

1. Copy the executable program `3dfs` into one of the directory of your $PATH (On Windows we recommend copying this file to the `C:\Windows` directory. Executable for common platforms are provided in the root directory of the CD:

   - Linux RedHat 7.0: `3dfs.redhat`
   - SGI's Irix 6.5: `3dfs.irix`
   - MS Windows 9x/NT/2000: `3dfs.exe`

   If your platform is not listed, the compilation of the `3dfs` program should be straightforward.

2. Copy the `data` directory located on the CD to your hard drive.

Step 1. Producing an FSC file

The 3D face space provided on this CD is formed by two distinctive subspaces: the 3D shape space and the 3D texture space. These two subspaces are each 49-dimensional. Hence a novel 3D face is defined by its 49 shape and texture coefficient. This input is provided to the `3dfs` program (see Step 2.) through an FSC file (FSC stands for Face Space Coefficients).

Alternative A - Using the file train.fsc provided on the 3DFS CD

The file `train.fsc` located in the root directory of the CD store the coefficients of the 50 training faces used to build the face space.

Alternative B - Producing your own fsc file to generate novel 3D faces

Generating novel 3D faces require the creation of a fsc file. Briefly, an fsc file is an ASCII file. Each one of its line defines a 3D face. The first word of a line is the obj file name where the 3D face is created. Then the shape and texture coefficients follows. More detailed description of the `FSC` file.

Example:

```
first_dim s 1 t 1
second_dim s 0 1 t 0 1
```

This `FSC` file generates 2 faces. The first one (stored in the file `first_dim.obj`) shows the impact of the first shape and texture dimensions and the second face shows the influence of the second shape and texture dimensions.

Step 2. Generation of 3D Faces as OBJ files

The program **3dfs** is used to generated 3D faces from an FSC file. These 3D faces are saved as OBJ files. Although the 3dfs program can be compiled on any machine since the source code (located in the directory 3dfs) is in the standard ANSI C language, executable for common platforms are provided in the root directory of the CD:

- Linux RedHat 7.0: 3dfs.redhat
- SGI's Irix 6.5: 3dfs.irix
- MS Windows 9x/NT/2000: 3dfs.exe

If your platform is not listed, the compilation of the 3dfs program should be straightforward.

To run the program 3dfs type the program name followed by the directory where the data are located and by the FSC file describing the faces to be generated (On Windows, this must be done is a DOS box). For instance, to generate the 50 faces used to train the face space, change the directory to a directory where you have write privileges (because the output faces will be stored in that directory) and type:

Linux RedHat 7.0 **3dfs.redhat -data_dir <dir> train.fsc**

SGI's Irix 6.5: **3dfs.irix -data_dir <dir> train.fsc**

MS Windows 9x/NT/2000: Click on the Start button, then follow Programs and click on **MS-DOS Prompt**. At the prompt change the directory where you would like the faces to be saved and type:
3dfs.exe -data_dir <dir> train.fsc

where <dir> is the path of the data directory.

ANNEX A: Using the 3D Face Space

Step 3. Viewing the 3D Faces (optional)

An OBJ file viewer is included on the 3DFS CD. To make it usable on many platforms, it was developed in JAVA. Hence it requires the *Java RunTime Environment* to be installed and the *JAVA 3D API*. Refer to the ObjViewer page for more details about the installation and the usage of the viewer.

CD Content

`3dfs/`	directory containing the source code (ANSI C) of the program **'3dfs'**
`objviewer/`	directory storing the Java source code of the program **'ObjViewer'**
`data/`	directory holding the shape and texture principal components
`3dfs_50.html`	HTML Documentation for the program **'3dfs'**
`objviewer.html`	HTML Documentation for the program **'ObjViewer'**
`3dfs.redhat`	Executable Program **'3dfs'** for REDHAT 7.0
`3dfs.irix`	Executable Program **'3dfs'** for IRIX 6.5 Systems
`3dfs.exe`	Executable Program **'3dfs'** for Windows 9x/NT/2000
`train.fsc`	Example input File for the program **'3dfs'**

Sami Romdhani
Last modified: Tue May 22 19:15:59 CEST 2001

ANNEX B: List of the 50 faces put into the Face Space

Names according to the data distribution of the University of South Florida

02341 f
03501 m
03503 m
03507 f
03513 m
03515 f
03516 m
03518 m
03519 f
03521 m
03523 m
03525 f
03526 m
03527 m
03541 m
03543 m
03545 f
03546 m
03548 f
03549 m
03552 m
03553 m
03554 m
03555 f
03556 m
03557 m
03559 f
03569 m
03585 m
03536 f
03575 m
03578 f
03580 m
03581 m
03586 m
03588 m
03590 f
03594 f
03596 m
03597 m
03598 m
03601 f
03603 m
03618 f
03628 m
03629 m
03634 m
03636 m
03643 m
03647 m