COOPERATIVE PATCH-BASED 3D SURFACE TRACKING

Martin Klaudiny, Adrian Hilton

Centre for Vision, Speech and Signal Processing, University of Surrey, UK
M.Klaudiny@surrey.ac.uk, A.Hilton@surrey.ac.uk

Abstract

This paper presents a novel dense motion capture technique which creates a temporally consistent mesh sequence from several calibrated and synchronised video sequences of a dynamic object. A surface patch model based on the topology of a user-specified reference mesh is employed to track the surface of the object over time. Multi-view 3D matching of surface patches using a novel cooperative minimisation approach provides initial motion estimates which are robust to large, rapid non-rigid changes of shape. A Laplacian deformation subsequently regularises the motion of the whole mesh using the weighted vertex displacements as soft constraints. An unregistered surface geometry independently reconstructed at each frame is incorporated as a shape prior to improve the quality of tracking. The method is evaluated in a challenging scenario of facial performance capture. Results demonstrate accurate tracking of fast, complex expressions over long sequences without use of markers or a pattern.

Keywords: dense motion capture, temporal alignment, surface tracking, cooperative optimisation

1 Introduction

Over the last decade, there has been an increasing research effort in spatio-temporal reconstruction of dynamic scenes using multi-view video acquisition. In particular, dense motion capture from several video streams has gained an interest for applications such as capturing full-body performance, face performance or cloth dynamics. The desired outcome of motion capture in these applications is a temporally consistent mesh sequence with a fixed topology which accurately models deformation of the observed surface over time. This enables use of real-world data in animation pipelines where it can be further modified or used as basis for new content.

The standard approach to the acquisition of a temporally consistent surface model combines 3D shape reconstruction with motion estimation by 2D optical flow. Initial work was presented by Vedula *et al.*[19, 20] where they introduce *scene flow* - a 3D vector field describing motion of a surface between two time frames. Scene flow is calculated by fusion of 2D flow vectors from individual views for each surface point on the volumetric model of an object. This concept was later modified to not rely on the precomputed 3D shape but works only for moving parts of a scene [21]. Further extensions focus on incorporating error statistics into the optical flow and disparity computation [11] or on integration of the scene flow with the shape reconstruction [24]. These techniques present an estimation of the scene flow between two frames on a regular grid across a scene. However, for applications such as performance capture the goal is to obtain motion of a 3D model of a subject over a longer period of time.

A possible solution for the tracking of non-rigid surfaces over relatively long sequences is temporal alignment of a sequence of unregistered geometries by deforming a template mesh according to 2D optical flow precomputed in each view. The motion of mesh vertices between frames is optimised jointly with respect to constraints given by the optical flows and raw geometries. The resulting scene flow is usually regularised by a deformation framework to suppress incorrect constraints. This type of framework has been demonstrated for a single camera by Hernadez *et al.*[9] and in a multi-camera setup by Zhang *et al.*[23]. Due to the use of frame-to-frame optical flow any errors in the tracking are propagated to the estimated 3D motion which leads to a drift of the mesh. Bradley *et al.*[3] address the drift by additional optical flow estimation in the UV domain of the mesh after the initial deformation. The residual flow is exploited to correct the final positions of the vertices. However, the results are not satisfactory in regions undergoing fast and complex motion.

Another group of methods estimates the scene flow directly without relying on 2D optical flow or precomputed raw geometry. These approaches mostly use a variational formulation of matching image information across views and over time. Pons *et al.*[15] alternate between multi-view stereo and scene flow in the same framework using a global image-based matching score. The shape of an object and its 3D motion are calculated frame-to-frame in the volumetric representation. The extension of this work [5] merges multi-view stereo and scene flow in a single energy functional and integrates it over a temporal window. Additionally, the complexity of task is increased by simultaneous optimisation across the whole mesh model. Several approaches address calculation of *a disparity flow* which is a reduced definition of the scene flow for the binocular case [13, 10, 22]. These approaches differ in a construction of the energy functional in the variational framework.

Previous techniques suffer from high computational complexity because of the integration across the whole surface or scene. A more tractable and flexible solution is offered by *3D tracking* approaches which estimate the scene

flow using small independent 3D elements attached to the surface. Carceroni and Kutulakos [4] propose a comprehensive model for the surface patches which consists of 3D position and orientation, curvature coefficients, diffuse/specular reflectance and linear transformation over time. This results in a complex optimisation scheme with a high number of parameters. A simpler approach [6] extends the Lucas-Kanade 2D tracking algorithm [1] to the 3D domain which leads to an alignment of a textured planar surfel with images from multiple cameras. The surfel has a single texture template with limited update to minimise the risk of drift. Neumann and Aloimonos [14] use a multi-resolution subdivision surface model instead of a collection of separate patches. A change of the subdivision model between frames is iteratively refined by shape+motion optimisation of individual surface patches.

Furukawa and Ponce [7] associate the patches with triangle fans around vertices of a mesh reconstructed by multi-view stereo in a reference frame. Thus, fixed reference textures attached to the patches shrink/stretch together with the deformed mesh. This allows improved alignment of patch appearance with multiple images during shape and motion optimisation between frames. After individual 3D tracking of patches with the aid of a simple motion expansion the locations of vertices are regularised by Laplacian smoothing combined with local mesh rigidity. This method captures fairly complex motions of patterned surfaces and is able to recover from errors and moderate occlusions. A rigidity term in the regularisation has to be relaxed in the tangent plane of the surface to accommodate extensive stretching and shrinking of materials such as human skin [8]. However, the approach fails with insufficient surface texture such as a face without patterned make-up.

Our work introduces a dense motion capture framework which overcomes the limitations of previous approaches (such as Furukawa and Ponce [7]) on weakly textured surfaces undergoing complex deformation. The inputs are multiple video sequences captured by synchronised and calibrated cameras, a sequence of unregistered geometries created by an arbitrary multi-view stereo method and a reference mesh providing a guideline for topology of the tracked full-resolution mesh. The output is a temporally registered sequence of meshes which accurately captures dynamics of the observed object.

A surface patch model described in Section 2 is built for the full-resolution mesh created from the reference mesh in the initial frame. A set of deformable surface patches associated with the mesh vertices is used to estimate their motion between successive frames. A novel approach to 3D patch matching over time aligns a multi-view texture from the previous frame with the current images from multiple views. The patch is forced to stay in the proximity of the provided raw geometry which limits gradual drift from the actual surface (Section 3.1). The main contribution is a cooperative optimisation scheme for the 3D matching of individual patches (Section 3.2). Their positions are iteratively optimised by interleaving local random sampling with the propagation of intermediate solutions among the neighbours. This scheme significantly increases robustness and accuracy of motion estimation in the case of rapid complex deformations of the surface. It also enables tracking of weakly textured surfaces such as skin. The estimated vertex displacements serve as soft constraints for a Laplacian deformation which regularises the motion of the entire mesh (Section 3.3). Effective suppression of outlying displacements is achieved by an original global constraint weighting based on the patch matching errors. This improves the motion regularisation and also retains efficient linear solution to the Laplacian deformation.

The proposed approach to surface tracking is evaluated on several datasets of complex facial performance with different levels of texture variation (dense random pattern, sparse markers and no make-up). Results demonstrate stable temporal consistency of the final mesh sequence with low drift over a long performance even for the face without any make-up (Section 5).

2 Surface patch model

A model of the observed surface is built according to the concept presented by Furukawa and Ponce [7]. The surface is represented as a triangular mesh $M = (V, E)$, where $V = \{\mathbf{v}_1, \mathbf{v}_2, ...\}$ is a set of 3D positions of the vertices in a defined world coordinate system (WCS). $E = \{(i, j), ...\}$ is a set of undirected edges among the vertices. A set of adjacent vertices around the vertex i is denoted $N_i = \{j|(i, j) \in E\}$. Every vertex has a respective surface patch associated with it which is shaped according to the adjacent triangle fan with the vertices in N_i. To represent a pose of the patch i independently from the mesh, each patch has its own local coordinate system (LCS) as depicted in Figure 1. A transformation between LCS and WCS is defined by a translation vector $\mathbf{p}_i$ and a rotation vector $\mathbf{r}_i$ in axis-angle representation. The patch pose is initially defined in a way that the origin of LCS has the position $\mathbf{v}_i$ of the respective vertex i. Z_L-axis is aligned with a vertex normal given by the surrounding faces. X_L, Y_L-axes are on the tangent plane such as $Y_L = Z_L \times X_W, X_L = Y_L \times Z_L$ where X_W is an axis of WCS.

The patch is covered with a grid of 3D sample points G_i which is centred around the vertex i (Figure 1). The sample points are placed on the neighbouring faces along O rings with growing radius from the central vertex. The corner samples of individual rings on the edges are spaced with a fixed distance d_o in 3D space. The value of d_o is chosen in a way that sample rings of any patch do not project further than 1 pixel apart in every view. This ensures correct sampling of image information without aliasing. The number of sample points increases with ring index o (central sample has $o = 1$). The ring o has $o - 2$ uniformly spaced points between the corner samples on every face. The sample grid can extend beyond N_i but it still follows the planes of triangle fan. The locations of samples are stored in barycentric coordinates with respect to the triangles they lie in. The benefits of this representation

are that the sample grid automatically changes shape with a modification of N_i and actual 3D positions G_i can be easily recomputed. The points G_i are expressed in LCS, so a change of $\mathbf{p}_i$ and $\mathbf{r}_i$ leads to the movement of the entire sample grid in a rigid manner. A visibility set C_i of the patch i contains views where the central vertex i is not occluded by other parts of mesh M and the vertex normal (Z_L-axis) points towards the cameras. The angle between the normal and flipped viewing direction of the camera is limited to 70° to avoid sampling artefacts when projecting a sample grid which is nearly perpendicular to an image plane.

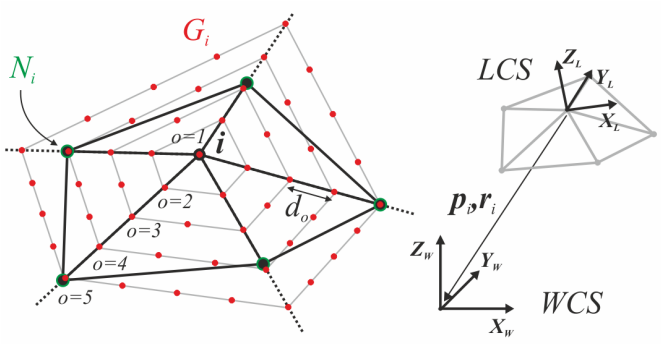

Figure 1: The patch related to the vertex i with the pose $\mathbf{p}_i, \mathbf{r}_i$. The sampling grid G_i has size $O = 5$ and spacing d_o. Its shape is given by 3D positions of the adjacent vertices from N_i.

Each patch has a multi-view texture B_i^c attached to it which models its current appearance in individual views c. B_i^c is obtained from images at the frame t given the current shape of mesh M. The sample grid G_i is converted to WCS using the transformation matrix T_i formed from $\mathbf{p}_i$, $\mathbf{r}_i$. Afterwards, a vector of pixel values is sampled from each view c using the sample grid: $B_i^c(t) = I_t^c(T_i G_i)$. Operation $I_t^c()$ encapsulates the projection of 3D points using intrinsic parameters of the camera c and the sampling of an image at the frame t. Grey-scale pixel values are bi-linearly interpolated for the projected sample points. The use of colour information does not bring significant benefit. The texture is obtained only from the views which are included in C_i for the frame t.

In the context of surface tracking the variables related to mesh vertices or corresponding patches such as $\mathbf{v}_i$ change over time. Thus, it is denoted as a function of time $\mathbf{v}_i(t)$ when required. For brevity of notation, note that $\mathbf{v}_i = \mathbf{v}_i(t)$ where t is the current frame.

3 Surface tracking

Estimation of surface motion between successive frames is solved by a combination of cooperative matching of independent surface patches and global Laplacian deformation of the mesh.

3.1 3D matching of surface patch

Finding a correspondence for the patch i between the frames $t-1$ and t is posed as a two-fold problem: the alignment of multi-view texture from the frame $t-1$ with the images from individual views at the frame t and the placement of a patch in close proximity to the raw unregistered geometry for the frame t. This problem is formulated as a joint optimisation task where the patch sample grid is rigidly translated in 3D space to satisfy both criteria in an optimal way.

Equation 1 defines an error function E_i which is minimised by altering the position $\mathbf{p}_i$ of the patch i. Initial value of $\mathbf{p}_i$ coincides with the position $\mathbf{v}_i(t-1)$ of the respective vertex in the previous frame. The rotation vector $\mathbf{r}_i$ forming the transformation T_i together with $\mathbf{p}_i$ has a fixed value according to the shape of M in $t-1$. $\mathbf{r}_i$ could be optimised together with $\mathbf{p}_i$, but there is a little gain in terms of precision and a risk of obtaining suboptimal solution increases.

$$E_i(\mathbf{p}_i) = \left(\frac{1}{|C_i|} \sum_{c \in C_i} \overline{NCC}(I_t^c(T_i G_i), B_i^c(t-1)) \right) + w_g \rho(\|\mathbf{p}_i - \mathbf{g}_i\|, \sigma_g) \quad (1)$$

The first term in Equation 1 represents the error in multi-view alignment of patch texture $B_i^c(t-1)$ with the current images from the cameras in C_i. The vectors of pixel values $I_t^c(T_i G_i)$ are obtained by projecting the sample grid G_i shifted by T_i to each view c. The values sampled at the frame t are compared to the texture template $B_i^c(t-1)$ using normalised cross-correlation (note that $\overline{NCC} = 1 - (NCC+1)/2$ is the inverted value which represents an error). The sum of matching errors is normalised by the size of visibility set.

The second term in Equation 1 forces the patch origin $\mathbf{p}_i$ to stay nearby the point $\mathbf{g}_i$ which is the closest point on the unregistered geometry. A view from C_i which the patch faces the most is selected according to the patch normal. $\mathbf{g}_i$ is an intersection of the ray between $\mathbf{p}_i$ and optical centre of the selected camera with the raw geometry. The distance between $\mathbf{p}_i$ and $\mathbf{g}_i$ is penalised by Tukey bi-weight error norm ρ defined in Equation 2 [23]. The penalty is uniform beyond a distance σ_g which makes E_i less affected by large outliers in the raw geometry.

$$\begin{aligned} \rho(x, \sigma_g) &= 3\frac{x^2}{\sigma_g^2} - 3\frac{x^4}{\sigma_g^4} + \frac{x^6}{\sigma_g^6}, |x| \le \sigma_g \\ \rho(x, \sigma_g) &= 1, |x| > \sigma_g \end{aligned} \quad (2)$$

This term effectively restricts the search space for the patch position. The key benefit is a limitation of the patch drift which otherwise leads to gradual degradation of the surface shape over time.

Both terms are linearly combined with the weighting coefficient w_g. A matching error $e_i = E_i(\mathbf{p}_i)$ cannot be evaluated for a particular value of $\mathbf{p}_i$ if $C_i = \emptyset$, $T_i G_i$ projects outside the image in any view from C_i or $\mathbf{g}_i$ cannot be computed because of missing data in the raw geometry.

3.2 Cooperative optimisation of patch 3D matching

The error function E_i has a complex behaviour in its parameter space. Therefore, minimisation of the patch i is likely to reach a suboptimal minimum if gradient descent or some other type of local search is employed (e.g. [7]). To tackle this problem we have extended the PatchMatch correspondence algorithm [2] to the domain of surface tracking. The concept of cooperative matching of individual image elements can be adapted for 3D surface patches. The essential assumption is that neighbouring patches on the surface move in a similar way. Thus, the patches can share intermediate outcomes of their own minimisations and increase the possibility of convergence to their individual global minima.

The optimisation for all patches across the mesh is performed iteratively and in cooperation. At first, initial values of matching errors e_i are calculated at the vertex positions $\mathbf{v}_i(t-1)$ using the image information from the frame t. The patches are then traversed one by one in h iterations and their positions $\mathbf{p}_i$ are modified from $\mathbf{v}_i(t-1)$ to decrease e_i. A new solution for $\mathbf{p}_i$ is computed in two subsequent stages at every iteration.

Propagation stage: The patch i tries to adopt the current motion estimates from the patches in N_i which have already been processed in the current iteration. A candidate for $\mathbf{p}_i$ is calculated by adding a displacement vector $\mathbf{p}_j - \mathbf{v}_j(t-1)$ from the neighbour j to the original position $\mathbf{v}_i(t-1)$. If the candidate has lower error than the current estimate $\mathbf{p}_i$, it is taken as a new solution. Thus, the neighbouring patches are encouraged to have similar motion but the assumption is not enforced explicitly. This is different to global spatial smoothness used in previous non-rigid surface tracking schemes where the single optimisation is performed simultaneously across the whole surface. An advantage of the propagation is the ability to correctly recover motion discontinuities between different regions of the surface such as surface folds.

Random sampling stage: A local search for minima is performed in the area around the current solution. New candidate positions are generated by random perturbation of $\mathbf{p}_i$: $\mathbf{p}_i + q_{max}\alpha^a\mathbf{u}$. $\mathbf{u}$ is a random 3D vector sampled from uniform distribution in the range $(-1, 1)$ which is scaled by the current search range size. The size of range exponentially decreases by ratio α ($\alpha \in (0, 1)$; $\alpha = 0.5$ in our experiments) with increasing integer exponent a. For each value of a a fixed number of candidate vectors (5 in our case) is generated which results in a cloud of samples with increasing density towards its centre at $\mathbf{p}_i$. The range of random sampling is limited by the maximal bound q_{max} and the increase of a from 0 is stopped when $q_{max}\alpha^a < q_{min}$. E_i is evaluated for each candidate and the one with the lowest error updates $\mathbf{p}_i$.

The change of $\mathbf{p}_i$ from $\mathbf{v}_i(t-1)$ by both stages throughout all iterations is limited by a bounding box around $\mathbf{v}_i(t-1)$ with a half size q_{lim}. This explicitly avoids motion estimates with magnitudes beyond possible motion between two frames. The matching of a patch is unsuccessful if E_i cannot be evaluated at $\mathbf{v}_i(t-1)$ and at any candidate position suggested during the minimisation. The order in which the patches are processed by the propagation and random sampling stage is given by two rules. The next patch is selected according to: 1. the highest number of already processed neighbours in the current iteration, 2. the most promising neighbour in terms of the current error. This ordering increases the impact of the propagation stage.

Interleaving the propagation and random sampling stage allows a patch to find a minimum of its own error function incrementally. The solution of each local search is challenged by the motion estimates from the adjacent patches which can lead to further improvement. This approach has better ability to avoid the convergence to local minima on E_i than independent gradient descent used in previous patch-based techniques. This greatly increases robustness to rapid non-rigid shape changes such as mouth opening. Iterative processing does not significantly increase computational load because the patches converge to nearly final solutions in a few iterations. The likelihood of reaching optimal outcome across the whole mesh improves with its density. Larger number of patches are more likely to converge to their optimal solutions and this information is propagated across the surface.

3.3 Weighted Laplacian deformation

The cooperative 3D matching of patches produces a raw motion field described by 3D displacement vectors $\mathbf{d}'_i = \mathbf{p}_i - \mathbf{v}_i(t-1)$. A Laplacian deformation framework [18] is employed to regularise this field because of outliers and its non-ideal continuity. The mesh deformation tries to preserve the shape of the mesh $M(t-1)$ subject to the weighted motion constraints $\mathbf{d}'_i$. The outcome is a new set of displacement vectors $\mathbf{d}_i$ which define the final $\mathbf{v}_i(t)$ for the current frame: $\mathbf{d}_i = \mathbf{v}_i(t) - \mathbf{v}_i(t-1)$. In contrast to Furukawa and Ponce [7], the motion of the surface with respect to the previous frame is regularised instead of a surface smoothing combined with approximate preservation of the shape from a reference frame. Also, the constraints are explicitly weighted according to their matching errors rather than repeatedly filtered between regularisations if they are outliers. This leads to a simpler formulation of the regularisation and more efficient solution.

The Laplacian deformation of the mesh is posed as a single optimisation problem across all vertices in contrast to the per-patch 3D matching. Equation 3 formulates the functional which is minimised with respect to the displacements $\mathbf{d}_i$. The problem is solved separately for each coordinate, so $\mathbf{d}[x]$ denotes a vector containing x coordinates of all $\mathbf{d}_i$ (similarly for y, z).

$$\underset{\mathbf{d}[x]}{\operatorname{argmin}} \quad s\|\tilde{L}\mathbf{d}[x]\|^2 + \|W(\mathbf{d}[x] - \mathbf{d}'[x])\|^2 \qquad (3)$$

The functional consists of the smoothing and constraint terms which are weighted against each other by a smoothness coefficient s. The smoothing term regularises the motion field across the mesh using a discrete Laplacian operator. The matrix $\tilde{L}$ stacks up in rows the weights of Laplacian operators

computed using $M(t-1)$ for each vertex [12]. In fact, $\tilde{L}$ represents a linear combination of bi-Laplacian and Laplacian operator: $\tilde{L} = ((1-k)L^2 + kL)$. This is used to model bending and stretching properties of the surface by changing coefficient k.

The constraint term incorporates raw displacements $\mathbf{d}'_i$ in the form of soft constraints which have varying influence expressed by a weight matrix W. The weight w_i of a particular displacement $\mathbf{d}'_i$ is entered in a corresponding place on the diagonal of W as $\sqrt{w_i}$. The constraint weight is derived from the matching error e_i associated with $\mathbf{d}'_i$ which reflects well the quality of motion estimate according to our experiments. The mapping between e_i and w_i described in Equation 4 is a declining linear ramp with a half-width δ_e centred around a error threshold t_e.

$$\begin{aligned} w_i &= 0, (e_i - t_e) > \delta_e \\ w_i &= 1, (e_i - t_e) < -\delta_e \\ w_i &= -\frac{1}{2\delta_e}(e_i - t_e) + \frac{1}{2}, |e_i - t_e| \leq \delta_e \end{aligned} \tag{4}$$

The number of constraints can be generally smaller than the overall number of vertices (*e.g.*a failure of patch matching due to full occlusion). For the vertices without motion estimate $\mathbf{d}'_i$ the corresponding positions in W and $\mathbf{d}'[x]$ contain zero entries. Their final displacements are then derived purely from the motion of the whole mesh.

Minimisation of the functional in Equation 3 leads to an over-determined linear system defined in Equation 5. Solving this system in least-squares manner for each coordinate separately provides an optimal motion field between frames.

$$\begin{bmatrix} \sqrt{s}\tilde{L} \\ W \end{bmatrix} \mathbf{d}[x] = \begin{bmatrix} \mathbf{0} \\ W\mathbf{d}'[x] \end{bmatrix} \tag{5}$$

4 Sequential processing

The proposed surface tracking is used to sequentially compute the motion of the observed surface represented by the mesh M starting from initial frame t_r. The inputs are multi-view video sequences and full calibration data of respective cameras. A sequence of temporally unregistered meshes can be generated by an arbitrary multi-view stereo technique [17]. However, the raw geometries should accurately model the instantaneous shape of the object because they constrain tracking of the mesh with desired topology. The topology is defined by a user-specified coarse reference mesh M' which is a good approximation of the surface shape at the initial frame t_r. M' is then uniformly subdivided multiple times to achieve the required mesh density (each face is split into 4 new faces). After each subdivision the mesh vertices are conformed to the raw geometry to refine the mesh shape. The subdivided full-resolution mesh M is deformed to match the dynamics of the observed object over time.

The surface patch model is built for M at the frame t_r as described in Section 2. The following steps are then repeated for each pair of successive frames $t-1, t$ starting from $t_r + 1$.

1. The correspondences are found at the frame t for all patches associated with M using the cooperative 3D matching (Sections 3.1, 3.2).
2. The patch displacements drive the Laplacian deformation of M (Section 3.3).
3. The patch poses are updated according to a new shape of M for the frame t. $\mathbf{p}_i$ is set to a new value of $\mathbf{v}_i$ and $\mathbf{r}_i$ is changed to align the Z_L-axis with a new normal at the vertex i and roughly preserve the previous direction of X_L, Y_L-axes.
4. The visibility of patches is re-evaluated with respect to the new M.
5. The sample grids of patches are recomputed to reflect a shape change of related triangle fans. Note that new positions G_i are expressed in the updated LCS. Although every patch is treated as a rigid element during 3D matching in a particular frame, the modification of sample grid over time improves matching the patch texture to the images in the case of non-rigid deformation of the surface.
6. The multi-view patch textures B_i^c are sampled from the images at t using the updated G_i and C_i. The update of patch textures in every frame brings additional flexibility in terms of modelling surface appearance on top of the deformation of sampling grid. The patch can adapt to small geometrical details varying extensively over time or illumination changes. These types of appearance changes cannot be modelled by a fixed texture template initialised in a reference frame even if the patches deforms correctly with the underlying surface.

After processing the whole sequence the result is a temporally consistent mesh sequence with the topology of M.

5 Results

Evaluation of the proposed method is conducted in the case of facial performance capture. The long-term dense motion capture of a face presents a number of challenges: rapid movements, complex non-rigid deformations and weak skin texture. Three similar performances containing various exaggerated emotions are used for testing purposes. They differ in level of difficulty in terms of surface texturedness: face painted with a random pattern (denoted as *Pattern*; 310 frames), face painted with a set of markers (*Markers*; 367 frames) and face with a plain skin (*Plain*; 310 frames). An actor was captured by 4 HD cameras arranged into two vertical stereo pairs on each side of the face. The cameras were synchronised and fully calibrated with respect to WCS. Image sequences with resolution 1920×1080 pixels were recorded in HD-SDI uncompressed $4:2:2$ format at $25fps$. The actor was illuminated by uniform white light.

The temporally unregistered sequence of meshes is reconstructed by a stereo technique based on graph cut [16]. Both stereo pairs provide a sequence of depth maps for each side of the face. The depth maps are fused per frame

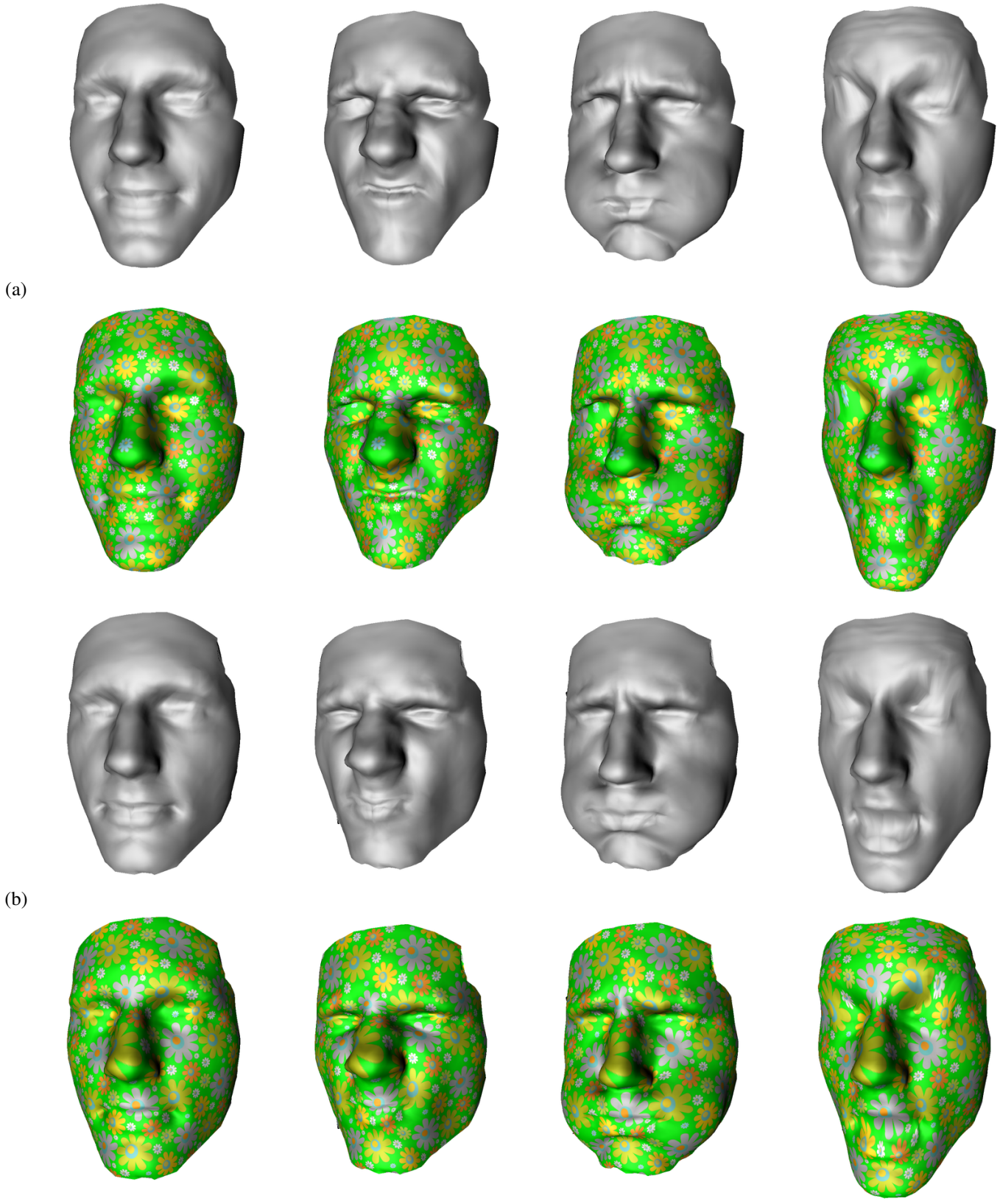

Figure 2: Snapshots from the temporally consistent mesh sequences created using the proposed method for the datasets *Pattern* (a), *Markers* (b). Rendered with a uniform material (top row), with a fixed UV texture attached in the initial frame (bottom row).

into single mesh with $\sim$ 21000 triangles. The facial shape is reconstructed up to skin folds and medium-sized wrinkles for all 3 types of data (shown in the supplementary video[1]). The surface jitters a bit over time and the magnitude of bumpiness increases from the well-textured surface (*Pattern*) to the least textured one (*Plain*). Outliers occasionally appear in problematic regions with view-dependent appearance such as eyes or inside of the mouth. However, this does not pose a problem for the surface tracking algorithm. The reference mesh M' was constructed manually for the first frame of each sequence with a topology designed according to the painted markers in the dataset *Markers* (328 triangles). M' is subdivided 3 times and conformed to a raw geometry to obtain the full-resolution mesh M (5248 triangles).

The following parameter sets are used as a baseline configuration of the algorithm in our experiments. All datasets share a subset of parameters: $O = 11, d_o = 0.2mm, \sigma_g = 10mm, h = 5, g_{min} = 0.1mm, k = 0.6, q_{lim} = 5mm$. The dataset-specific parameters are for *Pattern*: $w_g = 0.5, q_{max} = 2.5mm, s = 0.1, t_e = 0.05, \delta_e = 0.01$; for *Markers*: $w_g = 1.0, q_{max} = 1mm, s = 0.5, t_e = 0.15, \delta_e = 0.05$ and for *Plain*: $w_g = 1.0, q_{max} = 1mm, s = 1.0, t_e = 0.15, \delta_e = 0.05$. Computational time for a single frame is 1-2 minutes on 2.5 GHz processor using single-threaded C++ code (excluding reconstruction of the raw geometry).

5.1 Evaluation for different surface texturedness

The proposed surface tracking approach is evaluated in the context of surface texturedness on the datasets *Pattern*, *Markers* and *Plain* with varying amount of make-up. The key observation is that the surface of a face can be accurately tracked even without any additional make-up and the results are comparable to those with the aid of markers or a random pattern. Snapshots from the resulting temporally consistent mesh sequences are presented for *Pattern* in Figure 2(a), for *Markers* in Figure 2(b) and for *Plain* in Figure 3(a). The neutral expression at the beginning of every row is from the initial frame and the following expressions are in temporal order from the left to the right. We refer the reader to the supplementary video for playback over time and additional visualisations.

Figure 2(a) demonstrates correct capture of facial shape and its change over time for *Pattern*. The shape details such as wrinkles on the forehead are recovered with temporal consistency. The method is able to handle extensive surface deformations such as puffing out the cheeks or fast moving regions such as forehead and chin during a surprise. A very small drift can be observed over time in spite of a variety of rapid large motions. The unregistered mesh sequence contains finer details than the temporally consistent mesh sequence which can be noticed in the video. Some amount of detail is filtered out together with outliers during the weighting of deformation constraints. Also, continuous update of multi-view patch textures smooths surface detail to some extent.

The dataset *Markers* represents more difficult input, but the quality of temporal consistency is not compromised as shown in Figure 2(b). Accurate motion estimates from the strong features such as markers are successfully propagated to the skin regions among them. The dataset *Plain* poses the most challenging case which is not successfully addressed by the previous dense motion capture techniques. The strength of cooperative 3D patch matching manifests itself especially in this case where it enables to maintain temporal consistency over a long sequence of complex expression changes (Figure 3(a)). There is no significant drift of the mesh throughout the entire performance in spite of weak skin texture. However, a few local distortions appear in the eyes and inner lips because their appearance changes drastically over time. The drawback is a smoother shape of the mesh which can be observed for both markers and plain skin. This is caused by the stronger regularisation necessary to handle the lower quality of raw motion estimates in plain skin areas.

5.2 Comparison with independent gradient descent

To compare the proposed technique with previous patch-based techniques we implemented 3D patch tracking presented by Furukawa and Ponce [7] and combined it with our search space reduction by raw geometry and regularisation scheme. There are several main differences from the proposed approach in terms of the estimation of temporal correspondence. Firstly, motion of individual patches is independently optimised using a gradient descent. Secondly, the minimisation of error function E_i per patch is more complex and runs in two phases - normal components of a patch pose $(\mathbf{p}_i, \mathbf{r}_i)$ are minimised first and all components are refined together afterwards. Thirdly, the multi-view patch texture is initialised in the reference frame and stays fixed throughout the sequence. The results with independent gradient descent (IGD) are presented for *Pattern* in Figure 4(a), *Markers* in Figure 4(b) and *Plain* in Figure 3(b). The snapshots are taken at the same time instances as for the results by the proposed technique. Also, the same sets of parameters are used with IGD for individual datasets as with our method.

Direct comparison of the temporal consistency in the individual performances shows that IGD achieves plausible result only for *Pattern* (as reported in [7]). However, some local drift and distortion of the mesh occur during the most rapid emotions such as surprise (Figure 4(a)). The cooperative random sampling used in our method handles this situation correctly. The difference in performance between the methods is even more apparent for *Markers* and *Plain*. IGD does not cope well with tracking weakly textured patches and the mesh gradually degrades due to large drifts in both performances (Figures 4(b), 3(b)). In the video it can be seen that IGD fails during faster motions or large deformations of the skin and is not able to recover from the resulting mesh distortions. This showcases the robustness of cooperative patch 3D matching in such situations. Also, the update of patch multi-view texture

[1] Supplementary video is available under:
http://kahlan.eps.surrey.ac.uk/Personal/MartinKlaudiny/cvmp2011/index.html

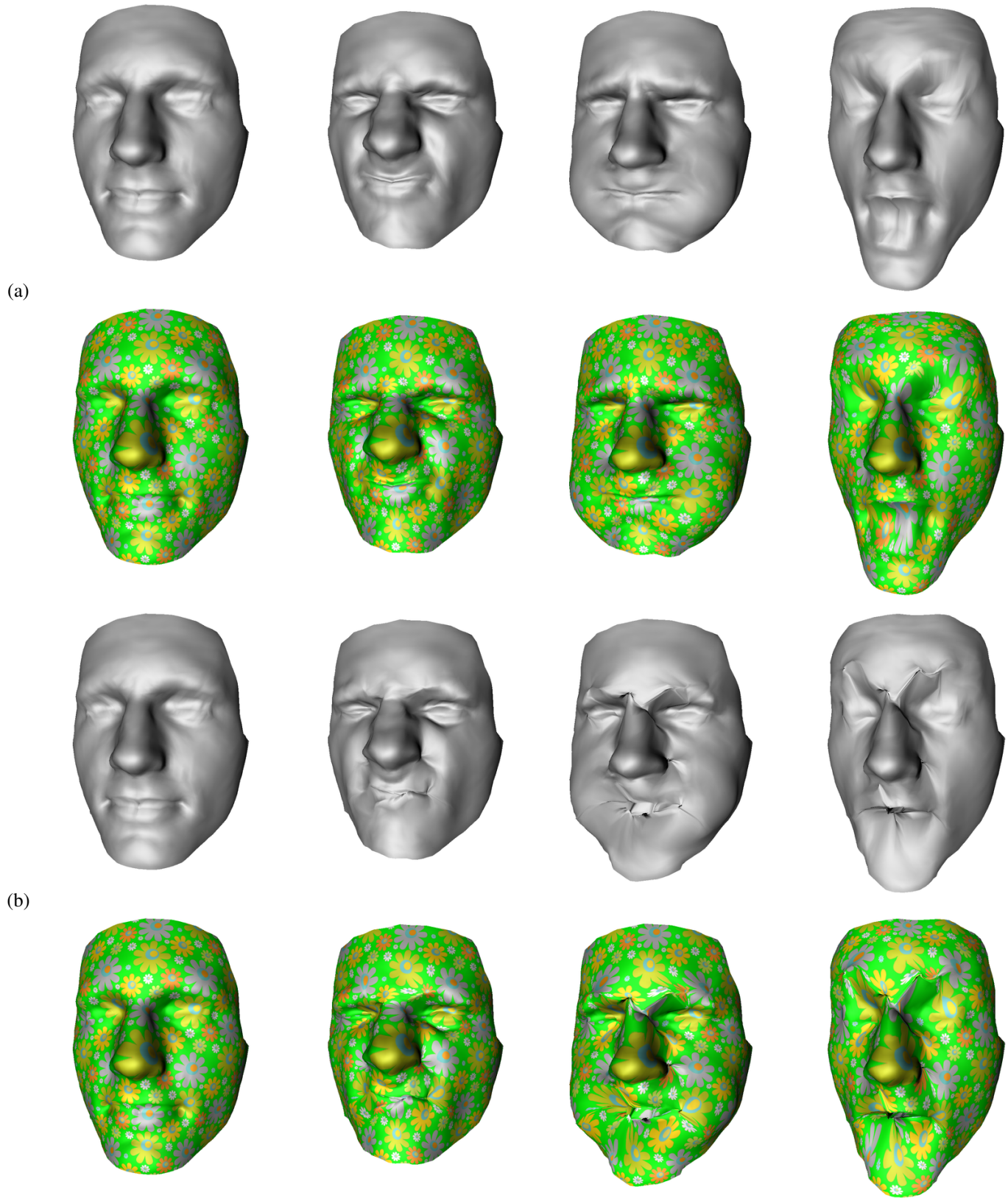

Figure 3: Snapshots from the temporally consistent mesh sequence created using the proposed method (a) and the independent gradient descent (b) for the dataset *Plain*.

in every frame brings benefits over the fixed texture template if the raw geometry prior is used. The adaptive texture is able to model extensive changes of skin appearance, thus allowing better patch matching throughout the entire performance. A potential drift over time due to the per-frame update of texture is limited by the constraint on a shape of the surface. A disadvantage is the mentioned partial smoothing of details which leads to a smoother facial model for *Pattern* than in the results presented in [7].

6 Conclusion

We have presented the novel dense motion capture method combining temporal 3D matching of deformable surface patches with the weighted Laplacian mesh deformation. The results in the scenario of facial performance capture demonstrate accurate tracking of large, rapid non-rigid surface deformations over long sequences. In comparison to the previous patch-based methods, the proposed approach significantly improves robustness in the case of rapid non-rigid motions and for surfaces with weak textures. This allows accurate motion capture of the face without the aid of markers or pattern which has not been successfully achieved by the state-of-the-art methods.

References

[1] S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework. *IJCV*, 56(1):221 – 255, March 2004.

[2] C. Barnes, E. Shechtman, D. Goldman, and A. Finkelstein. The generalized PatchMatch correspondence algorithm. In *ECCV*, September 2010.

[3] D. Bradley, W. Heidrich, T. Popa, and A. Sheffer. High Resolution Passive Facial Performance Capture. In *SIGGRAPH*, 2010.

[4] R.L. Carceroni and K.N. Kutulakos. Multi-view scene capture by surfel sampling: From video streams to non-rigid 3d motion, shape reflectance. In *ICCV*, volume 2, pages 60 – 67, 2001.

[5] J. Courchay, J.-P. Pons, P. Monasse, and R. Keriven. Dense and accurate spatio-temporal multi-view stereovision. In *ACCV*, 2009.

[6] F. Devernay, D. Mateus, and M. Guilbert. Multi-camera scene flow by tracking 3-d points and surfels. In *CVPR*, pages 2203–2212, 2006.

[7] Y. Furukawa and J. Ponce. Dense 3d motion capture from synchronized video streams. In *CVPR*, pages 1–8, 2008.

[8] Y. Furukawa and J. Ponce. Dense 3d motion capture for human faces. In *CVPR*, 2009.

[9] C. Hernández, G. Vogiatzis, G. J. Brostow, B. Stenger, and R. Cipolla. Non-rigid photometric stereo with colored lights. In *ICCV*, pages 1–8, 2007.

[10] F. Huguet and F. Devernay. A variational method for scene flow estimation from stereo sequences. In *ICCV*, 2007.

[11] R. Li and S. Sclaroff. Multi-scale 3d scene flow from binocular stereo sequences. *CVIU*, 110(1):75–90, 2008.

[12] M. Meyer, M. Desbrun, P. Schroder, and A. H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. *Visualization and Mathematics*, III:35–57, 2003.

[13] D. B. Min and K. Sohn. Edge-preserving simultaneous joint motion-disparity estimation. In *ICPR*, pages 74–77, 2006.

[14] J. Neumann and Y. Aloimonos. Spatio-temporal stereo using multi-resolution subdivision surfaces. *IJCV*, 47(1-3):181–193, 2002.

[15] J.-P. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *IJCV*, 72(2):179–193, 2007.

[16] S. Roy. Stereo without epipolar lines: A maximum-flow formulation. *IJCV*, 34:147 – 161, 1999.

[17] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, pages 519–528, 2006.

[18] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *SGP*, pages 175–184, 2004.

[19] S. Vedula, S. Baker, R. Collins, T. Kanade, and P. Rander. Three-dimensional scene flow. In *ICCV*, volume 27, page 722, 1999.

[20] S. Vedula, S. Baker, and T. Kanade. Image-based spatio-temporal modeling and view interpolation of dynamic events. *ACM TOG*, 24(1):240 – 261, 2005.

[21] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *TPAMI*, 27:475–480, 2005.

[22] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers. Efficient dense scene flow from sparse or dense stereo data. In *ECCV*, pages 739–751, 2008.

[23] L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM TOG*, 23:548–558, 2004.

[24] Y. Zhang and C. Kambhamettu. Integrated 3D scene flow and structure recovery from multiview image sequences. In *CVPR*, volume 2, pages 674–681, 2000.

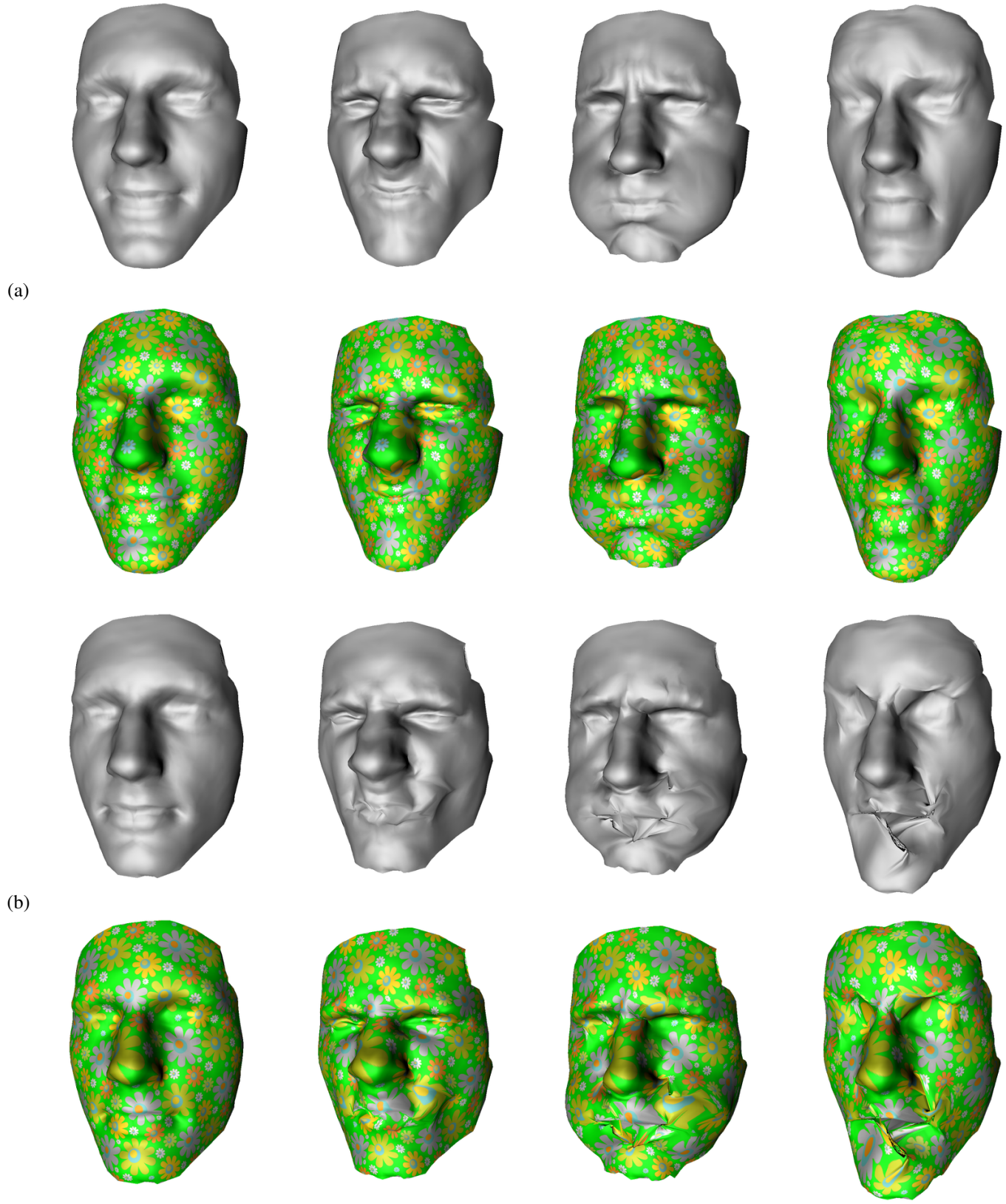

Figure 4: Snapshots from the temporally consistent mesh sequences created using the independent gradient descent for the dataset *Pattern* (a) and the dataset *Markers* (b).