# **Structure from Recurrent Motion: From Rigidity to Recurrency**

Xiu Li<sup>1,2</sup> Hongdong Li<sup>2,3</sup> Hanbyul Joo<sup>2</sup> Yebin Liu<sup>1</sup> Yaser Sheikh<sup>2</sup>

Tsinghua University<sup>1</sup> Carnegie Mellon University<sup>2</sup> Australian National University<sup>3</sup>

## Abstract

This paper proposes a new method for Non-Rigid Structure-from-Motion (NRSfM). Departing from the traditional idea of using linear low-order shape model for NRSfM, our method exploits the property of shape recurrence (i.e., many deforming shapes tend to repeat themselves in time). We show that recurrency is in fact a generalized rigidity. Based on this, we reduce NRSfM problems to rigid ones, provided that the recurrence condition is satisfied. Given such a reduction, standard rigid-SfM techniques can be applied directly (without any change) for the reconstruction of non-rigid dynamic shapes. To implement this idea as a practical method, this paper develops efficient algorithms for automatic recurrency detection, as well as for camera view clustering via a rigidity-check. Experiments on both synthetic sequences and real data demonstrate the effectiveness of the method. Since this paper gives a novel perspective on re-thinking structure-from-motion, we hope it will inspire other new researches in the field.

# 1. Introduction

Structure-from-Motion (SfM) has been a success story in computer vision. Given multiple images of a rigidly moving object, one is able to recover the 3D shape (structure) of the object as well as camera locations by using geometrical multi-view constraints. Recent research focus in SfM has been extended to the reconstruction of non-rigid dynamic objects or scenes from multiple images, leading to "Non-Rigid Structure from Motion" (or NRSfM in short).

Despite remarkable progresses made in NRSfM, existing methods suffer from serious limitations. Most notably is that they often assume some simple linear models, either over the non-rigid shape or over motion trajectories. These linear models, while are useful for characterizing certain classes of deformable objects (e.g, face, human pose, cloth), are unable to capture a variety of dynamic objects in rapid deformation, which are however common in reality.

This paper presents a new method for non-rigid structure from motion. Contrary to traditional wisdom for NRSfM, we do not make a linear model assumption. Instead, we describe how to exploit shape *recurrency* for the task of nonrigid reconstruction. Specifically, we observe that in our physical world many deforming objects (their shapes) tend to repeat themselves from time to time, or even only occasionally. In the context of SfM if a shape reoccurs in the video we say it is *recurrent*.

This observation of recurrence enables us to use existing knowledge of multi-view geometry to reconstruct the shape, if such a recurrence happens and is recognized. At first glance, this may be thought as a restrictive condition; however, to satisfy this condition is far easier than one thought. In fact, recurrent motions are ubiquitous in our surroundings, from human walking, to animal running, leaves waving, clock pendulum swaying, and to car wheels rotating etc. Many sport games like boxing and judo contain various repetitive motions. Having re-occurring movements is also an important design element used in dance choreography and gymnastics. Often, as long as a visual observation is long enough in time, revisiting a previously-seen scenario is highly probable.

Other merits of our method also include that: to recover the shapes of a non-rigid object, one can simply apply standard *Rigid-SfM* techniques, without having to develop new methods. For instance, rigid SfM techniques such as fundamental matrix computation, camera pose or PnP, rigid multi-view triangulation, bundle adjustment and rigid factorization can all be used at no change. Our method is suited to cases wherever shape seen in one frame repeats itself in another frame in time. We conducted experiments on synthetic data and real images. Both have validated the efficacy of our method.

# 2. The Key Insight

Rigidity is a fundamental property that underpins almost all works in rigid structure-from-motion. We say an object is *rigid* if its shape remains constant over time. For this reason, multiple images of the same object, taken from different viewpoints, can be viewed as redundant observations of the same target, making task of rigid SfM mathematically well-posed hence solvable. In contrast, the shape of



Figure 1. This composite slow-motion picture of 'figure-skating' clearly illustrates the basic idea of our non-rigid SfM method. Despite the skater's body poses kept changing dynamically over time, there were moments when she struck (nearly) identical posture, *e.g.* as indicated by the two red arrows and two blue arrows. Using a pair of such recurrent observations- albeit distant in time, one can reconstruct the 3D pose (shape) of the skater at that time instants, by using only standard rigid-SfM techniques.

a non-rigid object changes over time, violating the rigidity assumption and rendering NRSfM ill-posed.

In this paper, we show that *shape recurrency* is in fact a *generalized rigidity* in the following sense: we notice that many types of dynamic objects often repeat themselves at times. Given a video sequence, if one is able to recognize that a just-seen shape had been seen before, then these two instances of images can be used as a virtual stereo pair of the same *rigid* object in space; therefore any rigid-SfM technique can be applied to reconstruct its structure.

This is the key insight of the paper. To further illustrate this idea, consider the picture of Figure-Skating in Figure-1. The picture is a composite (strobe-type) photograph made by fusing multiple frames of slow-motion photos, which vividly captures the dynamic performance of the skater on ice. Examine carefully each of the individual postures of the skater at different time steps; it is not difficult for one to recognize several (nearly) repeated poses.

Despite conceptual our idea is, to actually implement it as a practical method requires novel and non-trivial (algorithmic) contributions. Specifically, in this paper we develop novel method to convert NRSfM problem to graphclustering problem solvable by Normalized-Cut. We will show how to quickly determine the probability that two images are projections of the same rigid shape, as well as how to achieve consistent reconstruction.

### 3. Problem Formulation and Main Algorithm

Consider a non-rigid dynamic 3D object observed by a moving pinhole camera, capturing N images at time steps of t = 1, 2, ..., N. Our task is then to recover all the N temporal shapes of the object, S(1), S(2), ..., S(N). To be precise, the shape of the object at time t,  $\mathbf{S}(t)$ , is defined by a set of M feature points (landmarks) on the object:  $S(t) = [X_{t1}, X_{t2}, ..., X_{tM}]$ , where  $X_{ti}$  denotes the homogeneous coordinates of the *i*-th feature point of the object at time t. Clearly S(t) is a  $4 \times M$  matrix.

Given a pinhole camera with projection matrix P, a single 3D point X will project to a point on the image at position x by a homogeneous equation  $x \simeq PX$ . For the shape of a temporally deforming object at time t, we have  $\mathbf{x}(t) \simeq P(t)S(t)$ , where  $\mathbf{x}(t)$  denotes the image measurement of the shape S(t) at time t, and P(t) defines the camera matrix of the t-th frame.

Collect all N frames of observations of the shapes of the non-rigid object, at time t = 1, ..., N, we obtain the basic equation system for N-view M-point NRSfM problem:

$$\begin{bmatrix} \mathbf{x}(1) \\ \mathbf{x}(2) \\ \dots \\ \mathbf{x}(N) \end{bmatrix} \simeq \begin{bmatrix} \mathsf{P}(1) & & \\ & \mathsf{P}(2) & \\ & & \dots & \\ & & & \mathsf{P}(N) \end{bmatrix} \cdot \begin{bmatrix} S(1) \\ S(2) \\ \dots \\ S(N) \end{bmatrix}$$
(1)

**Definition 3.1** (Rigidity). Given two 3D shapes, S and S', defined by their respective 3D landmark points in correspondence. We say they form a *rigid pair* if they are related by a rigid (Euclidean) transformation T. Note that a rigid transformation can be compactly represented by a  $4 \times 4$  matrix T, hence we have: S' = TS,  $\exists T \in SE(3)$ .

We use  $S \approx S'$  to denote that S and S' form a *rigid pair*.

**Example 3.1** (Rigid Object). The shape of a rigid object remains constant all the time:  $S(t) \approx S(t'), \forall t \neq t'$ .

**Example 3.2** (Periodic Deformation). A non-rigid object undergoing periodic deformation with period p will return to its previous shape after a multiplicity of periods, leading to  $S(t) \approx S(t + kp), \forall k \in \mathbb{N}$ .

**Example 3.3** (Recurrent Object). A shape at time t re-occurs after some  $\delta$ -time lapse:  $S(t) \approx S(t + \delta)$ .

### 3.1. Rigidity Check via Epipolar Geometry

If two 3D shapes (represented by point clouds) are given, checking whether or not they are rigidly related is a trivial task. However, this is not possible in the case of NRSfM where the shapes are not known *a prior*. All we have are two corresponding images of the shapes, and the rigidity-test has to be conducted based on input images only.

In this paper we use *epipolar-test* for this purpose. It is based on the well-known result of epipolar geometry: if two 3D shapes differ by only rigid Euclidean transformations, then, their two images must satisfy the *epipolar relationship*. Put it mathematically, we have  $S \approx S' \Rightarrow \mathbf{x'}_i^\top \mathbf{F} \mathbf{x}_i =$  $0, \forall i$ , where F is the *fundamental matrix* between the two images for S and S', respectively. Note that the RHS equation must be verified over all pairs of correspondences of  $(\mathbf{x}_i, \mathbf{x}'_i), \forall i$ .

Also note that satisfying epipolar relationship is only a *necessary condition* for two shapes S and S' to be rigid. This is because that the epipolar relationship is invariant to any  $4 \times 4$  projective transformation in 3-space. As a result, it is a weaker condition than the rigidity test, suggesting that even if two images pass the epipolar-test they still possibly be non-rigidly related. Fortunately, in practice, this turns out not to be a serious issue. This is because the *odds* that a generic dynamic object (with more than 5 landmark points) changes its shape precisely following a 15-DoF 3D projectivity is negligible. In other words, there is virtually no risk of mistaking.

The above idea of epipolar-test looks very simple. As such, one might be tempted to rush to implementing the following simple and straightforward algorithm:

- Estimate a fundamental matrix from the correspondences using the linear 8-point algorithm;
- Compute the mean residual error computed by averaging all the point-to-epipolar-line distances evaluated on key points in the image;
- 3. If this mean residual error is less than a pre-defined tolerance, return 'rigid', else return 'non-rigid'.

**Unfortunately**, despite the simplicity of the above algorithm, it is however not useful in practice. There are two reasons: (1)Ill-posed estimation: It is well known that linear methods for epipolar geometry estimation are very sensitive to outliers; a single outlier may destroy the fundamental matrix estimation. However, in our context, the situation is much worse (than merely having a few outliers). This is because, whenever the two feature point sets are in fact *not* rigidly related, forcing them to fit to a single fundamental matrix by using any linear algorithm can only yield a *meaningless* residual errors and unreliable decision. In short, fitting all feature points to a single epipolar geometry is ill-posed. Instead, in order to do a proper rigidity-test one must consider

the underlying 3D rigid-reconstructability of all these image points. (2) Degenerate cases: Even if two sets of points are indeed connected by a valid and meaningful fundamental matrix, there is no guarantee that a valid 3D reconstruction can be computed from the epipolar geometry. For example, when the camera is doing a pure rotation, there will not be enough disparity (parallax) in the correspondences to allow for a proper reconstruction–because the two cameras have only one center of projection- depth can not be observed. In such cases, the two sets of images can be mapped to each other by a planar homography, and the fundamental matrix estimations are non-unique.

**Our solution** is a new algorithm for rigidity-test, named "Modified Epipolar Test", which resolves both of the above issues: (1) it uses (minimum) sub-set sampling mechanism to ensure that the estimated two-view epipolar geometries (e.g., fundamental matrices) are meaningful; and (2) it adopts model-selection to exclude degenerate cases associate with planar homography. Detailed Epipolar-Test algorithm will be presented in Section-4.

### 3.2. Main Algorithm

Given the above rigidity-test is in place, we are now ready to present the main algorithm of the paper, namely *Structure-From-Recurrent-Motion (SFRM)*.

Algorithm	<b>n</b> 1: A high-level sketch of our Structure-	
From-Recurrent-Motion algorithm		
Innuts	N nonenactive views of a non-rigid shane	

	<b>input:</b> Iv perspective views of a non-fight shape
	S(t), t = 1,N. Choose K, i.e., the desired
	number of clusters.
	Output: The reconstructed 3D shapes of
	$S(t), \forall t \in \{1,, N\}$ up to non-rigid
	transformations.
1	for $(i = 1, \dots, N, j = 1, \dots, N)$ do
2	Call Algorithm 2 (i.e., modified-epipolar-test) to
	get A matrix whose $(i, j)$ -th entry $A(i, j)$ gives
	the probability that the two images $i, j$ are rigidly
	related.
3	end
4	[Clustering] Form a view-graph $G(V, E, P)$
	connecting all $N$ views, and the $A$ matrix is used as
	the affinity matrix. Run a suitable graph clustering
	algorithm to cluster the $N$ views into $K$ clusters.

5 [Reconstruction] Apply any rigid SfM-reconstruction method to each of the *K* clusters.

Note that the core steps of the algorithm are A-matrix computation and graph clustering. Note also, our algorithm only makes use of rigid SfM routines to achieve non-rigid shape reconstruction.

### 4. Modified Epipolar Test

In this section, we describe our modified epipolar-test algorithm. The output of this algorithm is the probability that these two images can be the projections of a same rigid shape. As discussed in the previous section, we do so by checking *whether or not these two sets of correspondences are related by a certain* fundamental matrix, *and at the same time* not *related by any planar homography*. The latter condition (i.e., excluding homography) is to ensure 3D reconstruction is possible. Our algorithm is inspired by an early work of McReynolds and Lowe for the same task of rigidity-checking [20], however ours is much simpler– without involving complicated parameter tuning and nonlinear refinement. Rigidity-checking was also applied for solving multi-view geometry problems without via camera motion [17].

We will proceed by presenting our algorithm description first, followed by necessary explanations and comments.

Algorithm 2: Modified Epipolar Test algorithm		
<b>Input:</b> Two input images, with M feature		
correspondences $\{(\mathbf{x}_i, \mathbf{x}'_i)   i = 1M\}$		
<b>Output:</b> The probability <i>P</i> that the two images are		
rigidly related.		

- 1. (Initialization): Set parameters  $\sigma_F, \sigma_H, \tau_F, \tau_H$ .
- 2. (Estimate fundamental matrices): Sample all possible 8-point subsets from the M points; Totally there are  $\binom{M}{8}$  such subsets. Store them in a list, and index its entries by *k*.

for  $k = 1, \cdots, \binom{M}{8}$  do

- Pick the k-th 8-point subset, estimate fund-matrix  $F_k$  with the linear 8-point algorithm.
- Given F<sub>k</sub>, compute the geometric (point-to-epipolarline) distances for all the M points by F<sub>k</sub>, *i.e.* d<sub>F</sub>(x'<sub>i</sub>, F<sub>k</sub>x<sub>i</sub>).
- Convert the distances to probability measures by applying Gaussian kernel. Compute the product of all probability measures by:

$$P_F(k) = \prod_{i=1..M} \exp\left(-\frac{d_F^2\left(\mathbf{x}'_i, F_k \mathbf{x}_i\right)}{\sigma_F^2}\right)$$
(2)

end

Find the minimum of all the  $\binom{M}{8}$  probabilities: *i.e.* 

$$P_F = \min_{k \in \binom{M}{8}} P_F(k). \tag{3}$$

### 3. (Estimate homography)

Run a similar procedure as above, for homography estimation, via sampling all 4-point subsets  $l \in \binom{M}{4}$ . The overall homography probability can be computed by:

$$P_H = \min_{l \in \binom{M}{4}} \prod_{i=1..M} \exp\left(-\frac{d_H^2\left(\mathbf{x}'_i, H_l \mathbf{x}_i\right)}{\sigma_H^2}\right) \quad (4)$$

4. (Compute overall probability) By now we have both  $P_F$ , and  $P_H$ . Compare them with their respective tolerances  $\delta_F$ , and  $\delta_H$ .

if  $(P_F \ge \tau_F)$  AND  $(P_H < \tau_H)$ , then Set  $P = P_F(1 - P_H)$ , return P. else Set P = 0, return P. end

### 4.1. Why does the algorithm work?

In Step-3 of the algorithm, we sample subsets of the data points, each consists of 8 points -minimally required to *linearly* fit a fundamental matrix. This way we avoid forcing to fit too many points to a single epipolar geometry. If the cameras are calibrated, one could also sample 5 points and use the non-linear 5-point essential-matrix algorithm for better sampling efficiency (c.f. [18]).

Once a fund-matrix  $F_k$  is estimated from an 8-tuple, we evaluate the probability of how likely every of the other feature points (not in the 8-tuple) satisfies this fund-matrix. Assuming all such probabilities are independent, the product of Eq.2 gives the total probability  $P_k$  of how well this  $F_k$ explains all the M points. Exhausting all  $\binom{M}{8}$  subsets, we pick the least one (in Eq.3) as a (*i.e.* conservative) estimate of the rigidity score. In Step-4, we repeat a similar sampling and fitting procedure for homography estimation. The idea is to perform *model-selection* [28] to filter out degenerate cases. Finally in Step-5, we report the overall probability (of rigidity-check for the two images) as the product of  $P_F$  and  $(1 - P_H)$  when  $P_F$  is sufficiently high  $(i.e. \geq \tau_F)$ and  $P_H$  is sufficiently low (*i.e.*  $< \tau_H$ ); otherwise report a '0'. In summary, our algorithm offers a way to estimate the rigidity-score as defined by the worst-case goodness-offitting achieved for all tentative fund-matrices for each 8tuple, while at the same time away from any homography.

#### 4.2. How to speed up the computation?

One might argue that, computationally, our Algorithm-2 is prohibitively expensive due to its exhaustive subset enumeration step (Step-3). For example, when M=100,  $\binom{M}{8}$  gives a large number of 186 billions.

Fortunately, below we will show that one can almost safely replace the enumeration step with a randomized sampling process with much fewer samples, yet at little loss of accuracy. Specifically, we only need to replace the first line (of "For  $k \in [1, \binom{M}{8}]$ ...") in Step-3 with "Randomly sample minimal 8-tuples for  $k \leq K$  times..".

Suppose we randomly sample a subset of 8 points from N points. Note the total number of combinations is  $\binom{N}{8}$ . Suppose there are about *e* proportion of valid subsets (*i.e. e* is the inlier ratio). By 'valid' we mean this 8-tuple gives rise to a good epipolar geometry which explains all data points well enough. Then the odds (*i.e.* probability) of picking a valid 8-tuple by only sampling once is *e.*, and the odds of getting an outlier is 1 - e. If one samples K times, then the total odds of getting at least one valid estimation is  $p = 1 - (1 - e)^K$ . As we will show next by using some numeric examples, this predicted odds can be very high in practice, suggesting that even a small number of random samples suffices. Note that this proof is akin to the probability calculation used in RANSAC.

# 5. View-Graph Clustering and Block-wise Reconstruction

For a given video sequence containing N views, we construct a complete view-graph G(V, E, A) of N nodes where each corresponds to one view. E denotes the set of edges, and A the affinity matrix in which A(i, j) measures the similarity between node-i and node-j.

After Step-3 of Algorithm-1, we have obtained an  $N \times N$  matrix P. We will use this P as the affinity matrix of our view-graph, *i.e.* A = P. Clearly, the picture of P provides a clear visualization that characterizes the dynamic movements of the object in the video. Bright colors in the matrix indicate at which views a particular shape re-occurred.

Figure-2 (from left to right) depicts for example three P matrices. From these pictures, one can easily discern that, the picture from left to right each corresponds to periodic motion, recurrent motion, and rigid motion respectively.



Figure 2. Examples of P matrices: (from left to right), periodic, recurrent, and rigid scenarios.

### 5.1. Spectral Clustering

Given a view-graph G(V, E, P) with the rigidity matrix P as its affinity matrix, and choose a suitable number K as the intended number of clusters, we suggest to use spectral

clustering technique to perform K-way camera view clustering. If two views are clustered to the same group, it means the two views are related by a rigid transformation.

Specifically, we use Shi-Malik's Normalized-cut for its simplicity. The algorithm goes as follows: First, compute a diagonal matrix whose diagonal entries are  $D(i,i) = \sum_j P(i,j)$ . Then, form a Symmetric normalized Laplacian by  $L = D^{-1/2}PD^{-1/2}$ . Next, take the least  $\log_2 K$  eigenvectors corresponding to the second smallest and higher eigen-values of L and run K-means algorithm on them to achieve K-way clustering. Some examples are given below.

**Example 5.1.** For periodic motion, K = 40 (*i.e.* one period):



Example 5.2. For general recurrent motion, K=25:



### 5.2. Block-wise Rigid Reconstruction

After the spectral clustering, the A matrix will be rearranged to a block-diagonal structure. Each block represents a cluster of views which are rigidly connected, up to an accuracy about the diameter of the cluster. Therefore, they can be considered as multiple rigid projections of the same shape. Hence any standard rigid-SfM technique can be used to recover the 3D shape. In our experiments specifically, we use incremental bundle adjustment which adds new frames gradually to a local triangulation thread.

#### **5.3. Scale Normalization**

As each cluster is reconstructed independently, all the shapes scale ambiguous. To achieve scale-consistent reconstruction results, we normalize the *etc*.

### 6. Results

The inputs to our method are multi-frame feature correspondences, like to many other NRSfM methods (*e.g.* [10, 1]). Finding feature correspondences is a difficult task in itself, especially for non-rigid deformable objects where self-occlusions may happen frequently. In our experiments, for synthetic data, the feature correspondences are naturally provided. For real data, specifically, we used the OpenPose [4] -a recently developed deep-net based landmark detection method– for sequences of human pose, face and hand. For other generic objects we used SIFT matching aided with manual correction.

#### 6.1. Periodic walking sequence

This first set of experiments aims to validate that our Algorithm-1 (and -2) works for real sequence with periodic movements – which is a special (and simpler) case of recurrent motion.

We use a person walking sequence in which the person walks at a constant speed, and a moving camera is observing him from different viewpoints, resulting in a nearly periodic sequence. We apply OpenPose[4] to detect and track 14 landmark points on the person over all 700 frames. Some sample frames are shown in Figure-3.

For the entire sequence, the rigidity (*i.e.* affinity) matrix computed by using our **Algorithm-2** is plot in Figure-4-Left. From this plot, it is clear that there exist strong periodicity, manifested as the bright bands along the main diagonals. Moreover the period can be readily read out as p=40 frames, despite our algorithm does not make use of this result. Instead, frames with repetitive shapes are automatically grouped together via view-graph clustering.

Figure-4-middle, and -right, each shows the re-arranged affinity matrix after spectral clustering, and the final clustering membership result, where the evident 'blocky' structure clearly reveals the grouping. We then perform a rigid-SfM for all views within each block. Figure-5 shows some example pose reconstruction results; note the poses are in 3D.

So far, our algorithm has only focused on recovering the non-rigid shape itself, ignoring its absolute pose in the world coordinate frame. In practice this however can be easily fixed, provided that the ego-motion of each camera view can be recovered by *e.g.* standard rigid-SfM/SLAM techniques against a stationary background. We conduct this experiment by first tracking background points, then estimating absolute camera poses relative to the background, followed by Procruste alignment between the absolute camera poses and each reconstructed human poses. A final sample reconstruction (with both background point clouds and human poses and trajectories) is given in Figure-6.



Figure 3. A (nearly) periodic walk sequence.



Figure 4. Affinity matrices before, and after spectral clustering (*i.e.* N-Cut). The 'blocky' structure becomes evident after N-cut. Right: the final view-clustering result.



Figure 5. 3D reconstruction results



Figure 6. Consistent 3D reconstruction of both dynamic foreground object (and temporal trajectories) and a static background scene.

#### 6.2. Recurrent dancing sequence

The aim for this second set of experiments is to demonstrate our new method's performance on a general (*nonperiodic*) video sequence which is likely to contain recurrent movements.

We choose a solo dancing sequence captured by the CMU Panoptic-Studio [14]. This dataset contains videos from multiple camera arrays. We extract a time-consecutive video from the dataset by randomly "hopping" between different cameras, to simulate a video as if captured by a "monocular camera randomly roaming in space".

This dancing sequence is challenging as the motion of the dancer is fast and the dance itself is complicated creating many unnatural body movements. For it, the computed affinity matrix is shown in Figure-7, clearly there is no obvious structure. However, after applying our graph-clustering, we can see clear block-wise pattern (albeit noisy), suggesting that the video indeed contains many recurrent (repetitive) body poses. Some example reconstruction results (along with the discovered recurrent frames) are shown in Figure-8.



Figure 7. The computed original affinity matrix, and the blockwise pattern after spectral cluttering on the CMU dancing sequence. There is no obvious cyclic pattern in the original affinity matrix. After graph clustering, more clear recurrence patterns are revealed.



Figure 8. 3D reconstruction results on the dance sequence.

### 6.3. Quantitative evaluation

To quantitatively measure the performance of our method, we use Blender to generate synthetic deformations with recurrence. We use flying cloth dataset [29] and fold the sequence by several times to mimic recurrency. Camera views are randomly generated. Figure-9 shows some sample frames of the data.

In this sequence, all ground-truth (object shape, camera poses) are given. Noises of different levels are added to image planes. Our method successfully detects recurrency and reconstructs the shape as shown in second row of Figure-9.

The reconstruction quality is measured by shape errors after alignment, as well as the portion of successfully reconstructed frames. We evaluate on two criteria at different noise levels. Results are given in Figure-11. Since they are self-explanatory, we omit further discussion here.

We compare our method with other state-of-the-art template-free NRSfM methods[1],[7]. The result is shown



Figure 9. Simulated waving cloth in wind and the recovered 3D shapes.

in Figure-10. In terms of overall reconstruction accuracy their performances are comparable, while ours is superior for frames exhibiting strong recurrency.



Figure 10. Comparison: Histograms of reprojection errors by different methods. Here we compare our method with [7] and [1].



Figure 11. SFRM performance at different noise levels. When noise increases, the reconstruction error increases whereas the success ratio falls. This result shows our method handles increasing amount of noises gracefully.

### 6.4. Timing

Figure-12 gives the timing results of our SFRM system (excluding rigid reconstruction), showing a clear linear relationship wrt. the number of feature points, as well as wrt. the number of random samples (in algorithm-2), but is quadratically related to the number of image frames. In our experiments we chose K -the number of clusters- empirically. For future work we would like to investigate how to automatically determine K.

We also test our method on face and hand data captured by the PanopticStudio. Sample results are shown in Figure-13. Since our used hand and face detectors produced noisy outputs, only visualizations are provided.



Figure 12. Timing (in seconds) as a function of #(random samples), #(points), and #( frames), respectively.



Figure 13. Sample 3D reconstructions on face and hand data.

### 7. Related work

The idea of our SFRM method is rather different from conventional NRSfM approaches. For space reason we will not review the NRSfM literature here but refer interested readers to recent publications on this topic and references therein ([7, 22, 15, 8]). Below, we focus on previous works with similar ideas.

A cornerstone of our method is the mechanism to detect shape recurrence in a video. Similar ideas had been proposed for periodic dynamic motion analysis [2, 25, 27, 26]. Our work was specifically inspired by [2, 12]. However, there are major differences. First, their methods assume strictly periodical motions, and need to estimate the period automatically [6] or manually [2]. This way, their methods can only handle limited periodical motions such as wellcontrolled walking and running. In contrast, our method extends to more general cases of recurrent motions, which include both *a-periodic*, and re-occurring cases, as well as rigid ones. Moreover, their methods assume static camera, and under the the periodical assumption, the target is not allowed to turn around and has to move (walking or running) on a straight line, capturing only partial surfaces [2] or trajectories [25]. Comparably, our method allows free-form target movements and camera motions. Finally, our method is fully automatic, while their methods rely on significant level of manual interactions.

Our method can be applied for 3D human pose recovery, therefore it is related to many works in this domain, [11, 24, 23, 21]. In particular, our method is of interest to those researches which try to lift 3D pose from 2D images, e.g. [3, 5, 19]. Earlier works in this direction either requires the integration of knowledge of the bone length of the target [16], or human pose and shape space priors [3]. Although in experiments we used 3D human poses, mainly as exemplar recurrent movements, our method does not take advantage of any category-specific priors. Rather, we treated poses as general point clouds in 3D. Another category of work on human pose capture relies on the existence of large-scale pose database for retrieving the most similar pose based on a 3D-2D pose similarity metric [9, 5, 13]. Their performance is heavily depend on the size and quality of the database of specific type of targets, while ours works in general scenarios. Recent deep learning approach by Martinez *et al.* [19] shows that a well-designed network for directly regressing 3D keypoint positions from 2D joint detection showed good performance. However, they rely on large amount of training data of specific class, while ours works without training.

### 8. Conclusion

We have presented a new method for solving Non-rigid Structure-from-Motion (NRSfM). It directly extends the concept of rigidity to recurrency as well as periodicity. With this new method at hand, one is able to directly use traditional rigid SfM techniques for non-rigid problems. Key contributions of this work include a randomized algorithm for robust two-view rigidity check, and a view-graph clustering mechanism which automatically discovers recurrent shape enabling the subsequent rigid reconstructions. Finite but adequate experiments have demonstrated the usefulness of the proposed method. The method is practically relevant, thanks to the ubiquity of recurrent motions in reality. One may criticize our method won not work if a shape is only seen once. We admit this is true but argue that it would be of little practical value to reconstruct any shape with such a fleeting nature. Our proposed view-graph and shape-clustering algorithms are examples of unsupervised machine-learning techniques. In this regard, we hope this paper may offer insights that bridge SfM research with machine learning.

Acknowledgement. We would like to thank reviewers for their comments. This work was completed when XL

was visiting PhD student to CMU under the CSC scholarship(No.201706210160). HL's research is funded by ARC Centre of Excellence for Robotic Vision (CE140100016). YL's research is funded by the National Key Foundation for Exploring Scientific Instrument No. 2013YQ140517, and the NSFC grant No.61522111

### References

- I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Nonrigid structure from motion in trajectory space. In *NIPS*, 2009. 5, 7
- [2] S. Belongie and J. Wills. Structure from periodic motion. In *Spatial Coherence for Visual Motion Analysis*, pages 16–24. Springer, 2006. 8
- [3] F. Bogo, A. Kanazawa, C. Lassner, P. V. Gehler, J. Romero, and M. J. Black. Keep it SMPL: automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016. 8
- [4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 6
- [5] C. Chen and D. Ramanan. 3d human pose estimation= 2d pose estimation + matching. In *CVPR*, 2017. 8
- [6] R. Cutler and L. S. Davis. Robust real-time periodic motion detection, analysis, and applications. *TPAMI*, 22(8):781–796, 2000.
- [7] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. *IJCV*, 107(2):101–122, 2014. 7, 8
- [8] M. Gallardo, T. Collins, A. Bartoli, and F. Mathias. Dense non-rigid structure-from-motion and shading with unknown albedos. In *CVPR*, pages 3884–3892, 2017. 8
- [9] A. Gupta, J. Martinez, J. J. Little, and R. J. Woodham. 3d pose from motion for cross-view action recognition via non-linear circulant temporal encoding. In *CVPR*, pages 2601–2608, 2014. 8
- [10] R. Hartley and R. Vidal. Perspective nonrigid shape and motion recovery. ECCV, pages 276–289, 2008. 5
- [11] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 2014. 8
- [12] P. P. Ivan Laptev, Serge Belongie and J. Wills. Periodic motion detection and segmentation via approximate sequence alignment. In *ICCV*. IEEE, 2013. 8
- [13] H. Jiang. 3d human pose reconstruction using millions of exemplars. In *ICPR*, pages 1674–1677, 2010. 8

- [14] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *ICCV*, 2015. 6
- [15] S. Kumar, Y. Dai, and H. Li. Multi-body non-rigid structure-from-motion. In *3DV*, 2016. 8
- [16] H. Lee and Z. Chen. Determination of 3d human body postures from a single view. *Computer Vision, Graphics, and Image Processing*, 30(2):148–168, 1985. 8
- [17] H. Li. Multi-view structure computation without explicitly estimating motion. In *CVPR*, pages 2777– 2784, 2010. 4
- [18] H. Li and R. Hartley. Five-point motion estimation made easy. In *ICPR*, ICPR '06, pages 630–633, Washington, DC, USA, 2006. IEEE Computer Society. 4
- [19] J. Martinez, R. Hossain, J. Romero, and J. J. Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, 2017. 8
- [20] D. P. McReynolds and D. G. Lowe. Rigidity checking of 3d point correspondences under perspective projection. *TPAMI*, 18(12):1174–1185, 1996. 4
- [21] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H. Seidel, W. Xu, D. Casas, and C. Theobalt. Vnect: real-time 3d human pose estimation with a single RGB camera. *TOG*, 2017. 8
- [22] H. S. Park, T. Shiratori, I. A. Matthews, and Y. Sheikh. 3d reconstruction of a moving point from a series of 2d projections. In *ECCV*, pages 158–171, 2010. 8
- [23] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis. Coarse-to-fine volumetric prediction for singleimage 3d human pose. *CoRR*, abs/1611.07828, 2016.
- [24] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3d human pose from 2d image landmarks. In ECCV, 2012. 8
- [25] E. Ribnick and N. Papanikolopoulos. 3d reconstruction of periodic motion from a single view. *IJCV*, 90(1):28–44, 2010. 8
- [26] S. M. Seitz and C. R. Dyer. Affine invariant detection of periodic motion. University of Wisconsin-Madison, Computer Sciences Department, 1994. 8
- [27] S. M. Seitz and C. R. Dyer. Detecting irregularities in cyclic motion. In *Motion of Non-Rigid and Articulated Objects, Proceedings of the 1994 IEEE Workshop on*, pages 178–185. IEEE, 1994. 8
- [28] P. H. Torr. An assessment of information criteria for motion model selection. In CVPR, pages 47–52. IEEE, 1997. 4
- [29] R. White, K. Crane, and D. A. Forsyth. Capturing and animating occluded cloth. In *TOG*, volume 26, page 34. ACM, 2007. 7