Solving Temporal Puzzles

Kshitij Adlakha
Stanford
kadlakha@stanford.edu
https://github.com/skitij/CS231AFinal

Abstract

Solving temporal puzzles is the problem of recovering the temporal order of a set of still images of a dynamic event, possibly taken by multiple uncaliberated camera. Photo-sequencing is an essential tool in analyzing (or visualizing) a dynamic scene captured by still images. The research paper that this project is based on, solves these temporal puzzles by describing the problem as a low order auto regressive model. The model can then be reduced to a mixed integer linear programming problem and can be solved with easily available solvers. The effectiveness and performance is evaluated and show to be generally applicable.

1. Introduction

The problem in photo sequencing is the process of recovering a temporal order from a set of distinctly captured images of a scene or event. These types of photo sequencing problems are frequently encountered in computer vision problems.

A common example of this is group photography, where a group of people are taking pictures of a live event, like a concert or a sports match. Images from group photography have been used commonly for analyzing the scene, for example 3D reconstruction of landmarks. However when these images have a temporal significance it can be useful to arrange them in time order to extract context.

The temporal sequencing problem is easier to solve if the 3D structure of the scene is known. In addition if the camera matrices (internal and external) are known the problem can be further simplified. However such information is not available for common group photography data-sets. The camera view points can change across images if a subject is being tracked, making the 3D structure extraction impossible. In addition people use a wide variety of cameras, and a single event could captured by a multitude of different cameras, making the problem of knowing camera matrix infeasible.

In the research paper that is used for this project, the authors have developed an algorithm for temporal sequencing that is widely applicable and makes few assumptions. The approach of ”the simplest approach is usually the right one” is applied, by assuming that for the dynamic model that is used describe the temporal sequence, the simplest model should be preferred. However if the time window is large, simpler models would not able not give effective results.

One application of temporal sequences that the paper discusses is for video encryption/decryption. The dataset was compiled from videos downloaded from YouTube, BBC Motion Gallery and datasets. The images in this dataset are taken by a single but fast moving camera and then they were shuffled, i.e. encrypted, in time. This dataset is challenging since these images are mostly dynamic as a whole, with few or no static objects in the field of view that could be used as a reference.

2. Related Work

The problem of photo-sequencing was initially introduced by Basha *et al.* [4]. The approach used by them is based on sampling the 3D locations along the point trajectory, at each of the time steps captured by the cameras.These 3D locations along the trajectory are an indicator of the temporal order of the images. They extract and match sets of static and dynamic features between each of the images and a reference image, without computing the 3D locations. Using the fundamental matrices computed by the static features, the corresponding dynamic features are projected onto the reference image. These projections can be used to infer the 3D trajectory of the scene point, and hence get the time ordering of images. Because of errors in measurement the dynamics features might not necessarily be consistent, so they use rank aggregation through Markov chain approximation.

In their followup papers Basha *et al.* [5] [6] tackle the space and time complexity of their algorithm. One of the restrictions with their earlier approach was that an image pair must be detected automatically for their 2D geometric based solution. In addition all feature points must appear and be matched to features in the static pair. This complicates the correspondence problem and limits the spatio-temporal extent of the event that can be captured. They address this problem by adding a new requirement that each camera capture takes more than a single photo, which gives them the temporal ordering across those pair of images. This simplifies the rank aggregation problem.

Despite the improvements in results, the assumptions made in those papers can be onerous. In contrast, in the paper by Dicle *et al*. [2] the definition of temporal puzzles is more general and is not restricted to event scenes, but it can also be applied to different video domains which can have dynamic textures and without a background static scene. In addition, their proposed approach does not require prior knowledge of partial ordering of the data and it can be applied to non-image sequences.

3. Technical Approach

3.1. LTI systems

A linear time-invariant system (or ”LTI system”) is a system that produces an output signal from any input signal subject to the constraints of linearity and time-invariance; Linear systems are systems whose outputs for a linear combination of inputs are the same as a linear combination of individual responses to those inputs. Time-invariant systems are systems where the output does not depend on when an input was applied. These properties make LTI systems easy to represent and understand graphically.

The transfer function of an LTI system is given by the Laplace transform of the impulse response of the system and it gives valuable information of the system's behavior and can greatly simplify the computation of the output response. the output of an LTI system will be given by the convolution of the signal with the impulse response. Since the convolution in the time domain is equivalent to a multiplication in the Laplace domain

$$\mathcal{G}(z) = \frac{Y(z)}{U(z)} = A.\frac{\prod_{i=1}^{m}(z - z_i)}{\prod_{i=1}^{n}(z - p_i)} = \sum_{i=1}^{n} \frac{\alpha_i z}{z - p_i}$$

The frequencies p_i and z_i which are the roots of the denominator and numerator of the transfer function are called the poles and zeros of the system, respectively. Poles and zeros are either real, or they must appear in complex conjugate pairs. Finally, bounded-input, bounded-output stable systems have all of their poles inside the unit circle.

3.2. Hankel matrices

A matrix whose entries along a parallel to the main anti-diagonal are equal, for each parallel. Equivalently, $H = h_{ij}$ is a Hankel matrix if and only if there exists a sequence s_1,s_2,$\dots$, such that h_{ij}=s_{i+j-1}, i,j=1,2,$\dots$. If s_k are square matrices, then H is referred to as a block Hankel matrix. Hankel matrices are frequently encountered in applications where the close interplay between polynomial and matrix computations is exploited in order to devise very effective numerical solution algorithms. The general form of a Hankel matrix is below.

$$A = \begin{bmatrix} a_0 & a_1 & a_2 & \dots & \dots & a_{n-1} \\ a_1 & a_2 & & & & \vdots \\ a_2 & & & & & \vdots \\ \vdots & & & & & a_{2n-4} \\ \vdots & & & & a_{2n-4} & a_{2n-3} \\ a_{n-1} & \dots & \dots & a_{2n-4} & a_{2n-3} & a_{2n-2} \end{bmatrix}.$$

Consider an nth order AR process: $y_{k+1} = \sum_{i=1}^{n} a_i y_{k-i}$. Given a set of N ordered noisy samples, possibly with missing data and corrupted with outliers, it is possible to estimate the underlying clean sequence $\{y\}_i$ by solving a structured rank minimization problem below where $p(y, d)$ is a data penalty term that depends on the missing data support and the noise-model.

$$\begin{aligned} \underset{\mathbf{y}}{\text{minimize}} \quad & \text{rank}\{\mathbf{H_y}\} \\ \text{subject to} \quad & p(\mathbf{y}, \mathbf{d}) \leq \eta_{max} \end{aligned}$$

3.3. Atomic Norm

Let $\mathcal{A}$ be a collection of atoms that is a compact subset of the set of real numbers. The elements of A are the extreme points of conv($\mathcal{A}$). Let $\|x\|_A$ denote the gauge of A.

$$\|\mathbf{x}\|_{\mathcal{A}} = \inf\{t > 0 \ : \ x \in t \ \text{conv}(\mathcal{A})\}.$$

Note that the gauge is always a convex, extended-real valued function for any set A. By convention this function evaluates to $+\infty$ if x does not lie in the affine hull of conv($\mathcal{A}$). It can be assumed without loss of generality that the centroid of conv($\mathcal{A}$) is at the origin. With this assumption the gauge function can be rewritten as

$$\|\mathbf{x}\|_{\mathcal{A}} = \inf\left\{\sum_{\mathbf{a}\in\mathcal{A}} c_{\mathbf{a}} \ : \ \mathbf{x} = \sum_{\mathbf{a}\in\mathcal{A}} c_{\mathbf{a}}\mathbf{a}, \ \ c_{\mathbf{a}} \geq 0 \ \forall \mathbf{a} \in \mathcal{A}\right\}.$$

If A is centrally symmetric about the origin then we have that $\|.\|_A$ is a norm, which is called the atomic norm induced by A.

For a convex penalty function given a set of atoms, [1] propose a convex optimization method to recover a “simple” model given limited linear measurements. Suppose that x is formed from a set of atoms and that we have a known linear map $\phi \ : \ R_p \rightarrow R_n$, and we have linear information about x as follows:

$$\mathrm{y} = \Phi \mathrm{x}^{\star}.$$

If the goal is to reconstruct x given y the following convex formulation can be used to accomplish this task :

$$\begin{aligned} \hat{\mathrm{x}} = \arg\min_{\mathrm{x}} \quad & \|\mathrm{x}\|_{\mathcal{A}} \\ \text{s.t.} \quad & \mathrm{y} = \Phi \mathrm{x}. \end{aligned}$$

When A is the set of one-sparse atoms this problem reduces to standard L1 norm minimization.

3.4. System Identification

The findings of [3] allow for identifying the system using the atomic norm.

In many applications, the true model can be decomposed as a linear combination of very simple building blocks. For instance, sparse vectors can be written as short linear combinations of vectors from some discrete dictionary and low-rank matrices can be written as a sum of a few rank-one factors. Previously, Chandraskearan *et al.* proposed a universal heuristic for constructing regularizers based on such prior information. If it is assumed that

$$G_{\star} = \sum_{i=1}^{r} c_i a_i \,, \text{ for some } a_i \in \mathcal{A}, c_i \in \mathbb{C}\,,$$

where A is an origin-symmetric set of "atoms" normalized to have unit norm and r is relatively small, then the appropriate penalty function is the gauge function (or the Minkowski functional) induced by the atomic set A.

$$\|G\|_{\mathcal{A}} := \inf\{t \;:\; G \in t \operatorname{conv}(\mathcal{A})\} = \inf\left\{\sum_{a\in\mathcal{A}} |c_a| \;:\; G = \sum_{a\in\mathcal{A}} c_a a\right\}$$

To apply these atomic norm techniques to system identification, we must first determine the appropriate set of atoms. For discrete time LTI systems with small McMillan degree, we can always decompose any finite dimensional, strictly proper system G(z) as:

$$G(z) = \sum_{i=1}^{s} \frac{c_i}{z - a_i}\,.$$

Hence, low order dynamical models can be estimated from experimental data by solving a problem of the form of the equation above to minimize the number of poles needed.

Minimizing the atomic norm in this equations above is an infinite dimensional, convex problem. To circumvent this obstacle, [3] proposed the Discretized Atomic Soft Thresholding (DAST) algorithm that uses an ϵ-net discretization of the unit disk in the complex plane, hence approximating the infinite dimensional set of first order stable LTI systems (atoms) by a finite one.

3.5. Algorithm

The algorithm to solve the puzzle is defined by the set of equations below :

$$\begin{aligned} \text{minimize}_{\mathbf{c}} \quad & \|\mathbf{c}\|_{\ell_1} \\ \text{subject to} \quad & \mathbf{v} = \mathbf{D_a c} \\ & \|\mathbf{Pu} - \mathbf{v}\|_{\infty} \leq \eta_{max}, \;\; \mathbf{P} \in \mathcal{P} \end{aligned}$$

here 'c' is the set of coefficients form the LTI transfer equation function. D_a is a dictionary created from the impulse response of atoms in the system. P is a permutation picked from the set of all possible permutations, and "u" is random shuffle order of the images. The goal would be to find P that can re-arrange the shuffled order back into the original order.

One constraint that is applied on the shuffling is that the first frame is kept in place, so we can add an extra constraint to the solver that the first frame in "u" should come before every other frame once P is applied. The overall algorithm is described below

Algorithm 1 Algorithm for temporal puzzles

```
Input: $\mathbf{S}$ dynamic sequence, $\mathbf{D_a}$ atoms dictionary, $\mathbf{Q}$ partial orderings, $D$ number of principal components,
Output: Permutation $\sigma$
Project $\mathbf{S}$ on $D$ principal comp., $\mathbf{u}_d \leftarrow \mathrm{PCA}_{D,d}(\mathbf{S})$
Convert $\mathbf{Q}$ to permutation constraints, $\mathbf{P_i}^T\mathbf{1} < \mathbf{P_j}^T\mathbf{1}$
Solve equation (15) with derivative $\dot{\mathbf{v}}_d = \mathbf{D_a c}_d$
```

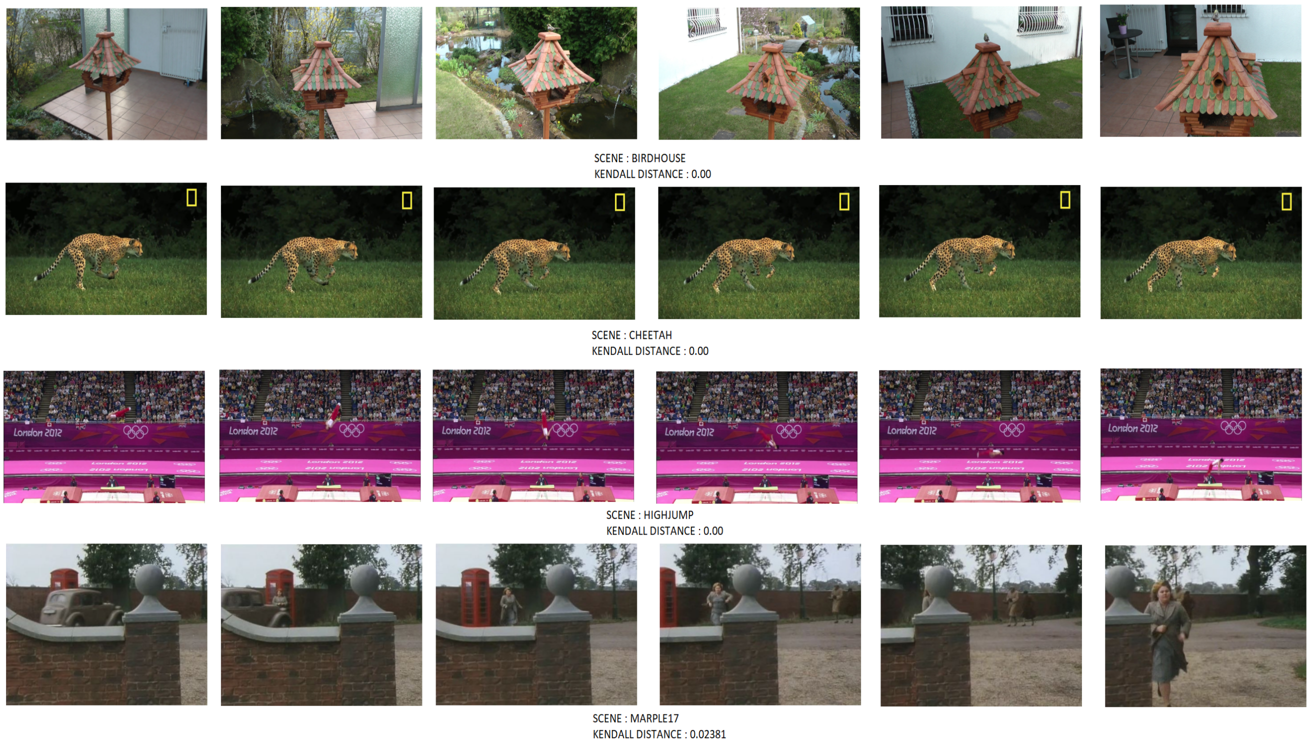
For the pole atoms in Da, the authors of [2] observed that the ring defined by $0.98 \leq |p| \leq 1.02$, where p belongs to the unit circle, with a discretization of $\epsilon = 0.05$ performs the best, so those same values are used with Dictionary size of 200 columns.

4. Experiments

A variety of image sequences were used for analyzing the algorithm. The accuracy of the results were measured using Kendall distance. The Kendall tau distance is a metric that counts the number of pairwise disagreements between two lists. The larger the distance, the more dissimilar the two lists are. Kendall tau distance is also called bubble-sort distance since it is equivalent to the number of swaps that the bubble sort algorithm would make to place one list in the same order as the other list.

For each of scenes run, the program is run for 10 iterations and average Kendall distance is reported. A summary of the Kendall distance for all the scenes are in the chart below.

The algorithm does very well for most scenes, as can be seen from the Kendall distance, even when the background is continuously changing between frames (birdhouse scene). However when the subject is stationary for



SCENE : BIRDHOUSE
KENDALL DISTANCE : 0.00

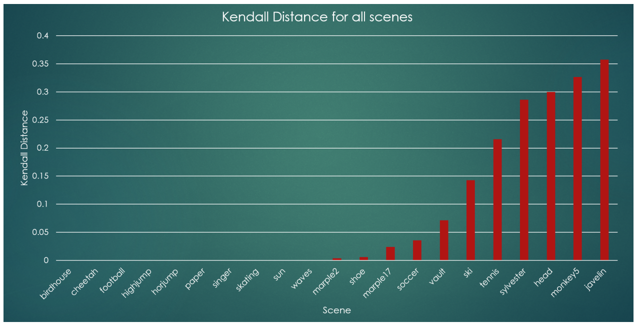
SCENE : CHEETAH
KENDALL DISTANCE : 0.00

SCENE : HIGHJUMP
KENDALL DISTANCE : 0.00

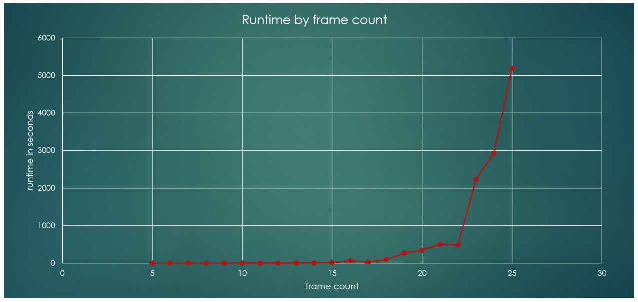
SCENE : MARPLE17
KENDALL DISTANCE : 0.02381

Figure 1. Data-set results

multiple frames, it is hard for the algorithm to find the correct ordering (monkey5 scene).



In terms of performance there is an exponential increase in run time when we cross 22 frames as input. SO the algorithm only looks to be well suited for smaller data set, in terms of frame count and time window. A chart of the algorithm performance can be seen below.



5. Conclusions

The [2] paper introduced a novel approach to solve temporal puzzles by use of the atomic norm framework. By applying the findings of [1] they were able to express an intractable mixed SDP problem in a mixed linear integer problem that can be solved using off the shelf constraint solvers.

As seen from the experiment results in the previous sections, there are scenes where the algorithm does very well, despite some scenes having a constantly varying background, or without a clear definition of a movement pattern (something that the previous approaches to this problem were not able to handle). At the same time there are scenes where the algorithm struggles. This can be seen in scenes that have repetitive motion, or where the subject is stationary for multiple frames. But these are hard cases, and other existing algorithms don't have a good way of handling them either.

In terms of performance, the algorithm is able to handle less than 10-15 fairly easily. But once the number of frames crosses 25, there is an exponential increase in run time, making the algorithm impractical. For large frame count the input will likely need to be batched. Also as the time window grows, the simplicity assumption does not hold, leading to wrong solutions.

References

[1] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12(6):805–849, Oct 2012.

[2] C. Dicle, B. Yilmaz, O. Camps, and M. Sznaier. Solving temporal puzzles. pages 5896–5905, 06 2016.

[3] P. Shah, B. N. Bhaskar, G. Tang, and B. Recht. Linear system identification via atomic norm regularization, 2012.

[4] Y. M. T. Basha and S. Avidan. Photo sequencing, 2012. In Computer Vision–ECCV 2012, pages 654–667.

[5] Y. M. T. Basha and S. Avidan. Space-time tradeoffs in photo sequencing, 2013. In Computer Vision (ICCV), 2013 IEEE International Conference on, pages 977–984. IEEE, 2013.

[6] Y. M. T. Basha and S. Avidan. Photo sequencing, 2014. International Journal of Computer Vision, pages 1–15, 2014.