AudeoSynth: Music-Driven Video Montage

Liao et al. SIGGRAPH 2015

Get a taste of it!

Presentation outline

- Motivation
- Previous work
- Problem formulation
- Definition of video and music segment
- Challenges
- Analysis (video + music)
- Synthesis (Energy Terms)
- Results

[Icon made by Icon Works from www.flaticon.com]

Motivation

Why do it at all?

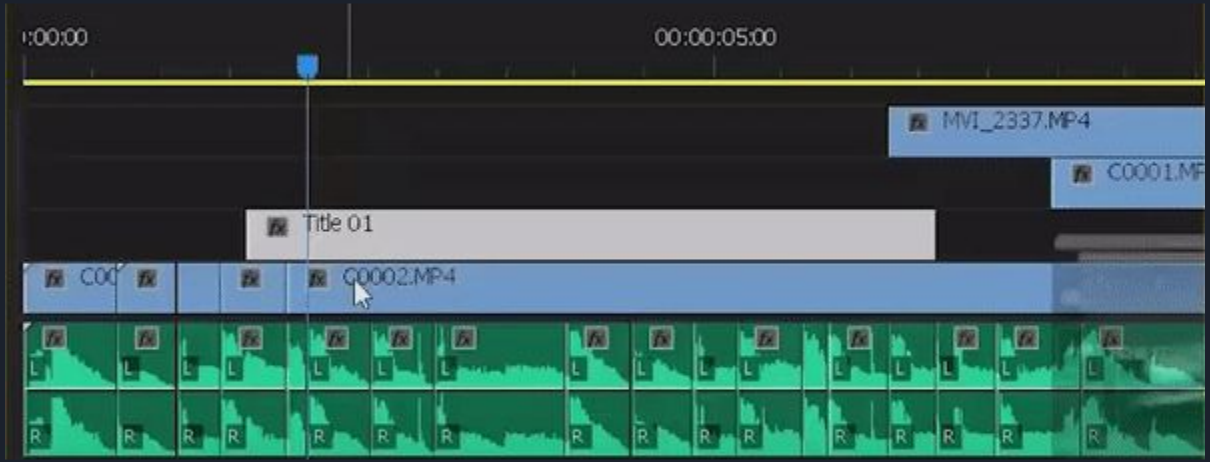
- Aesthetically compelling to match video content with the beats of music

Why do it automatically?

- Manually editing video to match a piece of music is very time consuming
- The composition has a large degree of freedom

Manuall mess

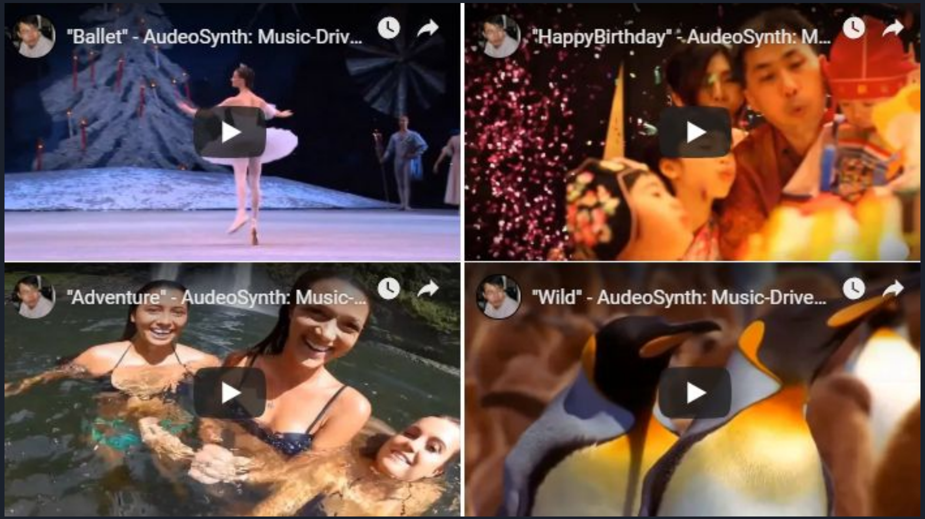
“so this is done by hand, it's just your hand touch **- listening to the specific piece of music you have** **over and over** and **kind of visualizing in your head the pacing of it and the beats per minute**. Whether it sounds slow or fast to you, but you could use these basic waveforms and cut and arrange things and place them on the beat to create a nice syncopated cut or cinematic sequence..”



Applications

Event aftermovies, adventure, sport and travel videos etc ..

(lets watch later)



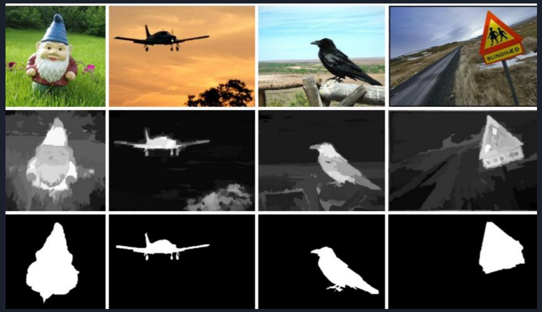
Related work

Music-driven imagery ..

Adapted solutions from:

- Optical flow [Liu et al. 2005].
  (*Motion magnification*)

- Saliency estimation [Cheng et al. 2014]
  (*Global contrast based salient region detection*)



Recall Visual Rhythm and Beat (Davis et al.)

Rhythm..

Visual beats..

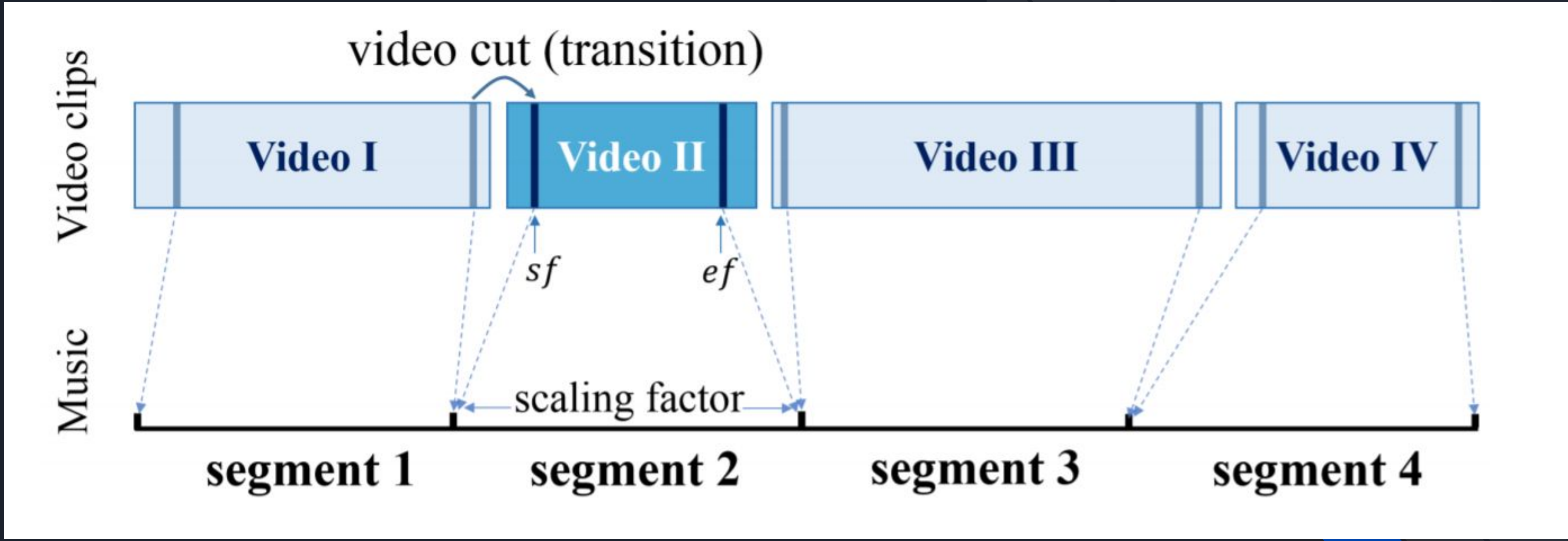
Saliency..

Will be revisited - keep in mind

Problem formulation

Match a video subsequence to each music segment

Match a video subsequence to each music segment



Match a video subsequence to each music segment

Essentials:

- Audio stays the same
- Play speed of video clips can be changed

Challenges

Remember the 3 challenges mentioned in the paper?

- Large degree of freedom
- Different types of media
- Large search space

Challenge #1

Large degree of freedom

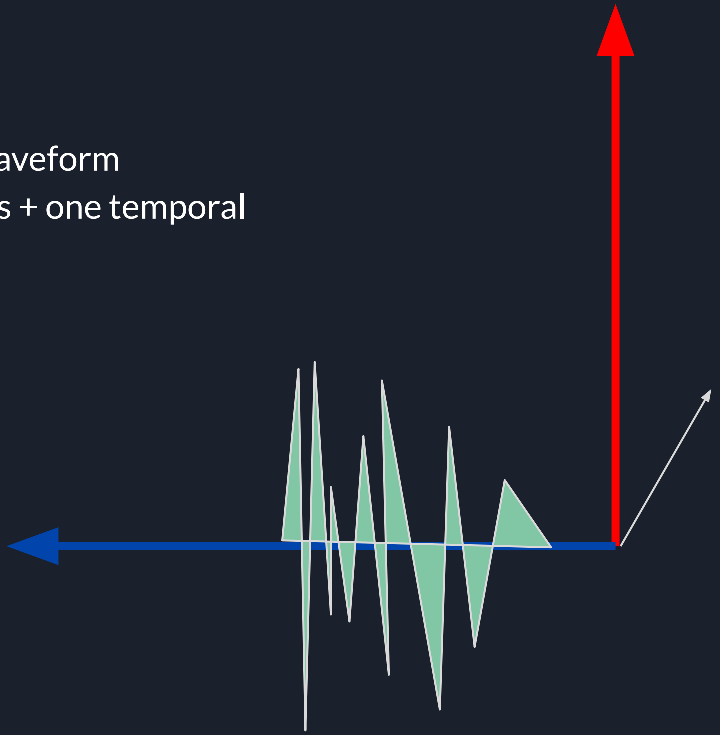
- which video clips do we want to use?
- when to cut?
- playback speed?

[image from unsplash.com]

Challenge #2

Different types of media

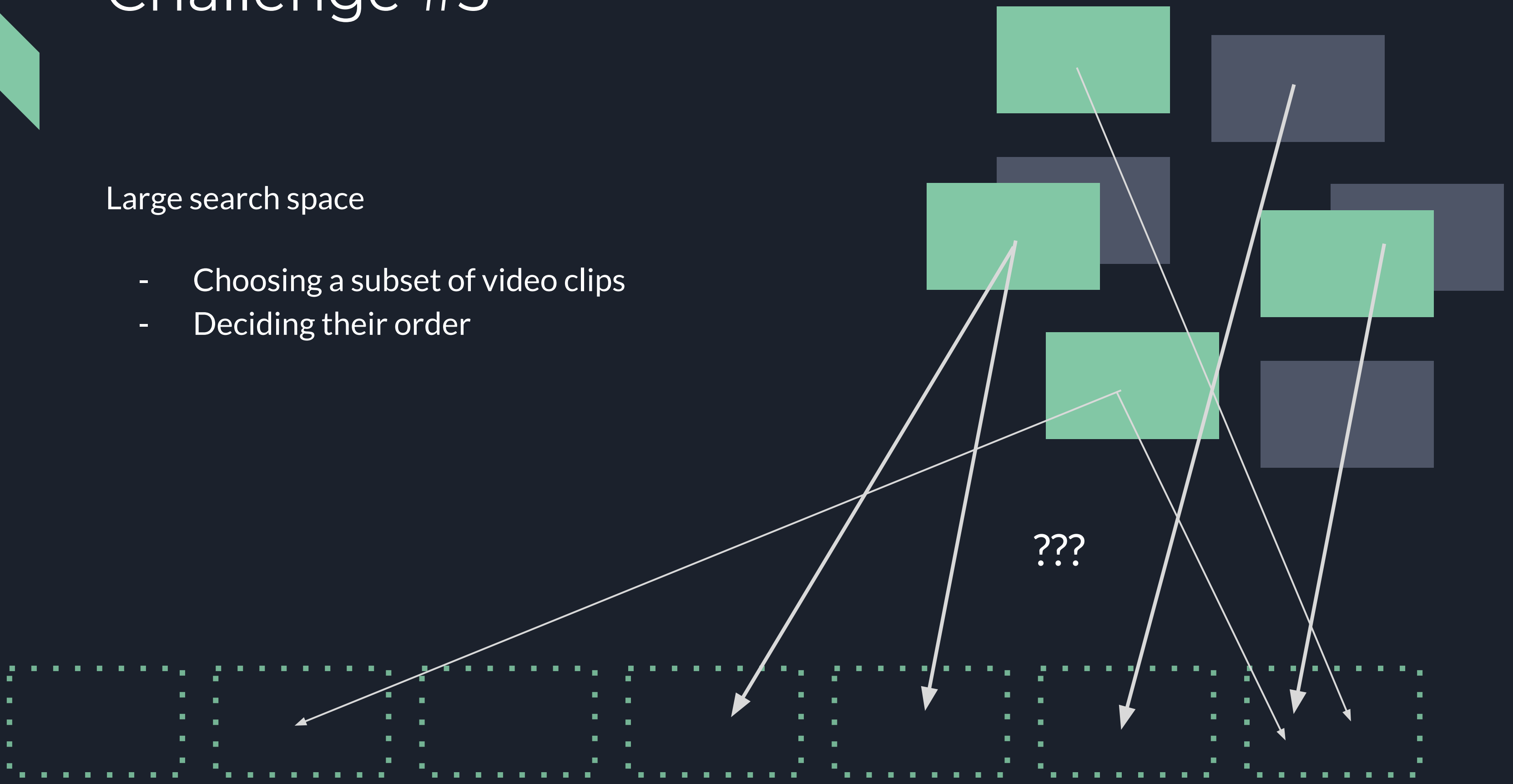
- Sound: one-dimensional in waveform
- Video: two spatial dimensions + one temporal



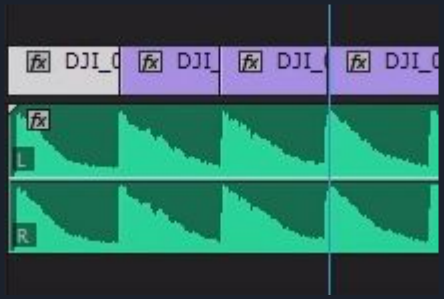
Challenge #3

Large search space

- Choosing a subset of video clips
- Deciding their order



Tackle the challenges

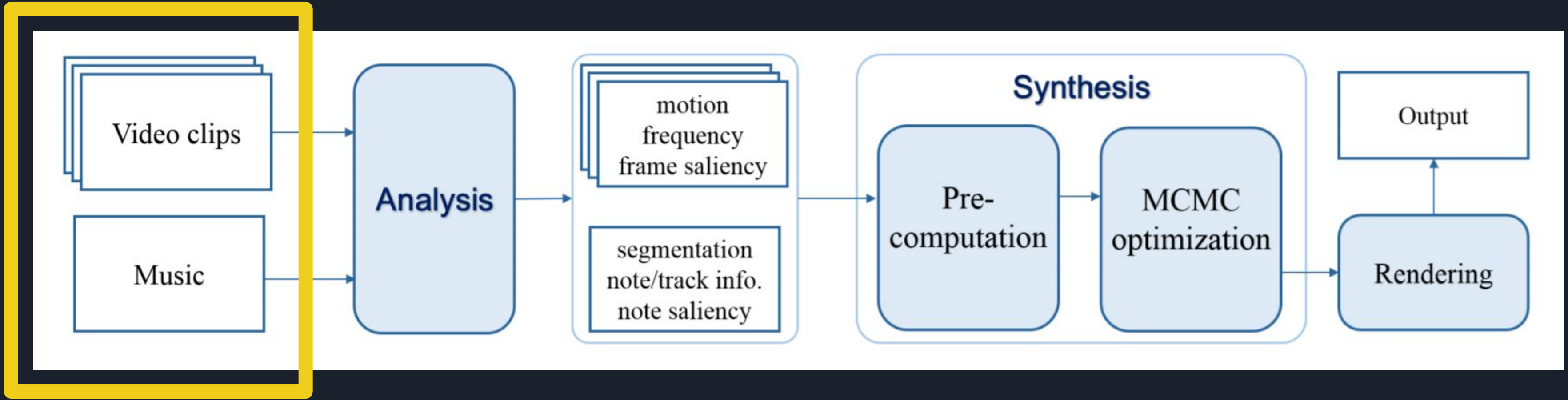


Narrowing down to two thumb-of-rules

- Cut-to-the-beat
- Synchronization
    - Extract features

[image from unsplash.com]

System overview



Problem formulation - A closer look

Match a video subsequence to each music segment

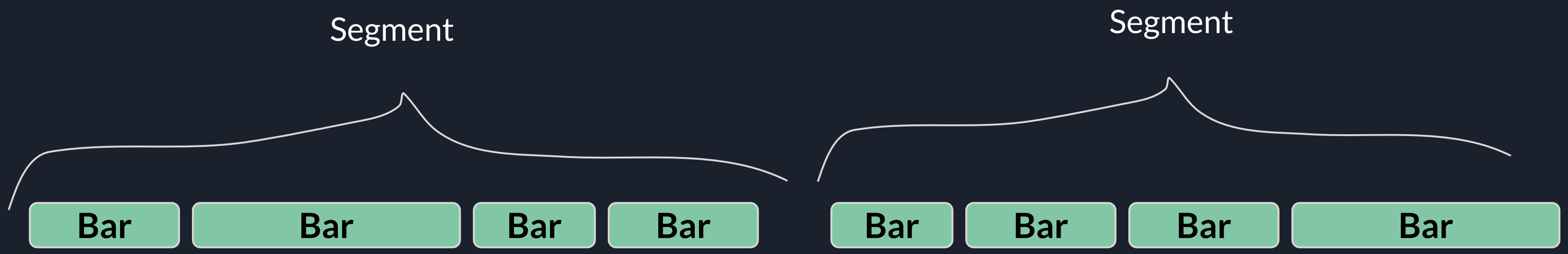
Before we even start thinking about the matching..

- How to define a video subsequence?
- And how to define a music segment?

Definition of a music segment

According to “cut to the beat” - Every music segment must start with a bar

Where bar is “the most basic unit of a music piece” in the MIDI format



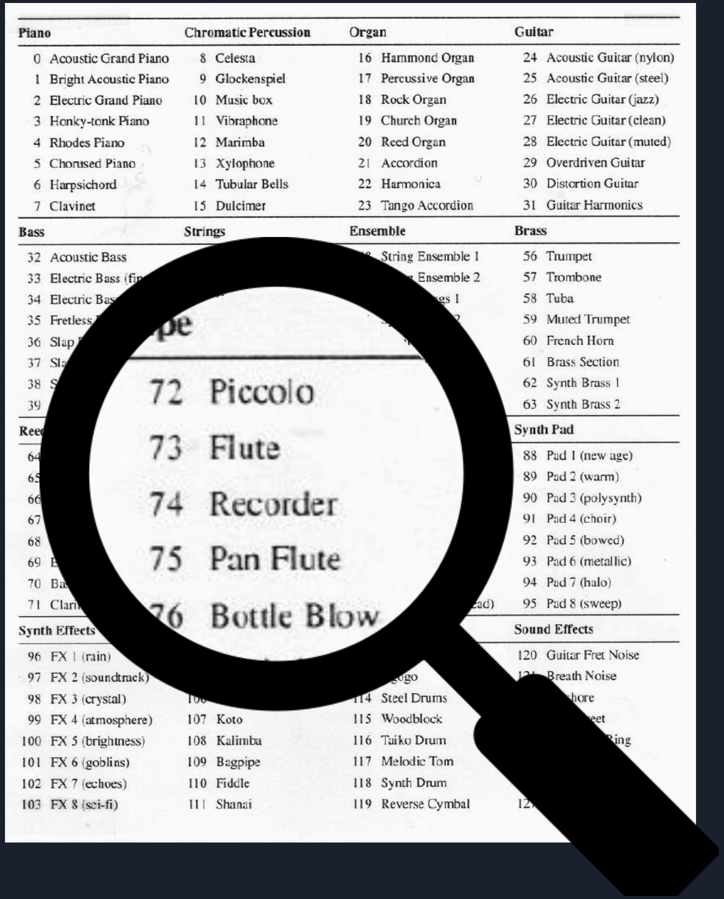
MIDI format

An encoding of musical signals
MIDI data: Sequences of musical note events

- Specifying note onset parameters:

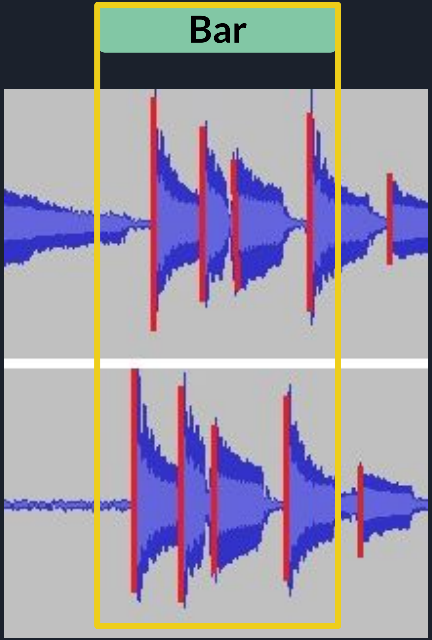
  - time
  - pitch
  - volume
  - duration

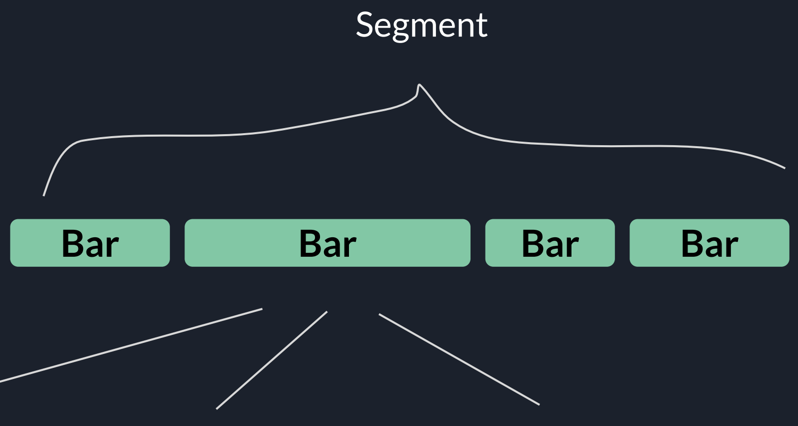
Why not waveform or mp3?



[MIDI sheet from http://www.cs.uccs.edu/~cs525/midi/midi.html]

MIDI format





track 0
Time: 2.5 seconds
Instrument: Piano
Volume: 80
Pitch: 50

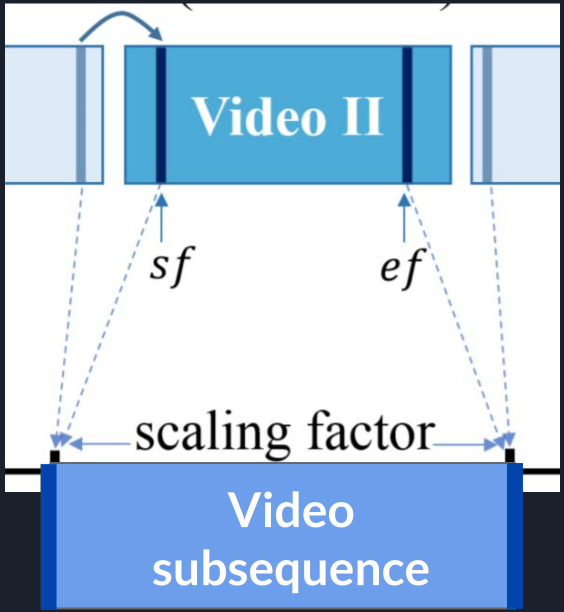
track 1
Time: 3.0 seconds
Instrument: Flute
Volume: 60
Pitch: 40

track 2
Time: 1.3 seconds
Instrument: Violin
Volume: 50
Pitch: 70

Definition of a video subsequence

Giving a video clip, the video subsequence is determined by:

- the start frame **sf**
- end frame **ef**
- scaling factor **scale**



Now we're ready for the Energy function!

Initial video clips: $\mathbf{V} = \{v_1, v_2, ..., v_p\}$

Sequential segments of input music: $\mathbf{M} = \{m_1, m_2, ..., m_q\}$

$$E(\boldsymbol{\theta}, \mathrm{M}) = E_{match}(\boldsymbol{\theta}, \mathrm{M}) + E_{transit}(\boldsymbol{\theta}, \mathrm{M}) + E_{global}(\boldsymbol{\theta}, \mathrm{M}), \quad (1)$$

Unknown parameters: $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_q\},$

$$\theta_i = (v_{a_i}; sf_i, ef_i, scale_i)$$

What is θ_i ?

Solution to the energy minimization:

$$\mathbf{V} = \{v_1, v_2, ..., v_p\} \quad \mathbf{M} = \{m_1, m_2, ..., m_q\} \quad \boldsymbol{\theta} = \{\theta_1, \ldots, \theta_q\}$$

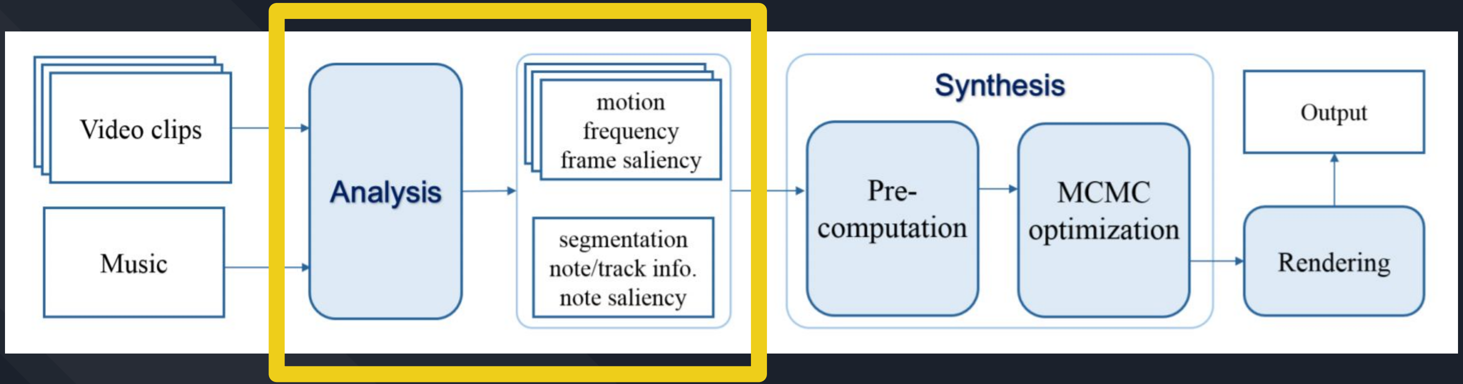
$$E(\boldsymbol{\theta}, \mathbf{M}) = E_{match}(\boldsymbol{\theta}, \mathbf{M}) + E_{transit}(\boldsymbol{\theta}, \mathbf{M}) + E_{global}(\boldsymbol{\theta}, \mathbf{M}), \tag{1}$$

a mapping function, $a : i \rightarrow j$

.. that maps each music segment $i \; (= 1, \ldots, q)$

.. to a subsequence of a video clip $j \; (\in \{1, \ldots, p\})$

Analysis


Video clips
Music
Analysis
motion
frequency
frame saliency
segmentation
note/track info.
note saliency
Synthesis
Pre-
computation
MCMC
optimization
Output
Rendering

Video Analysis

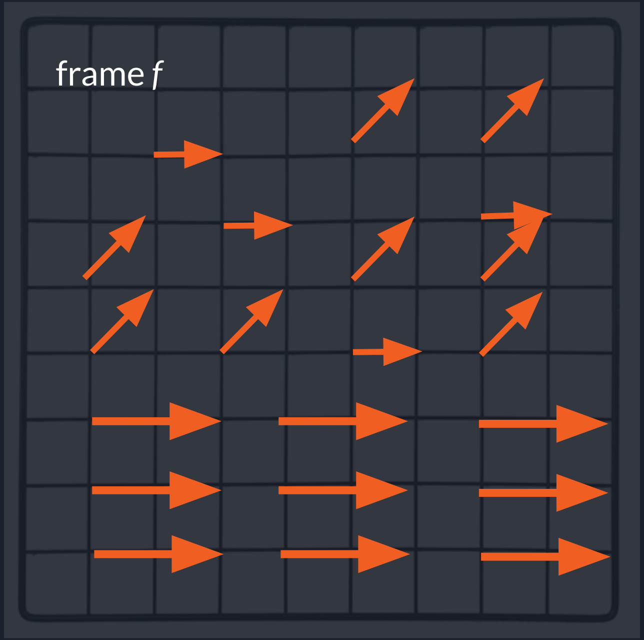
What to we need to know to make a good match with a music segment?

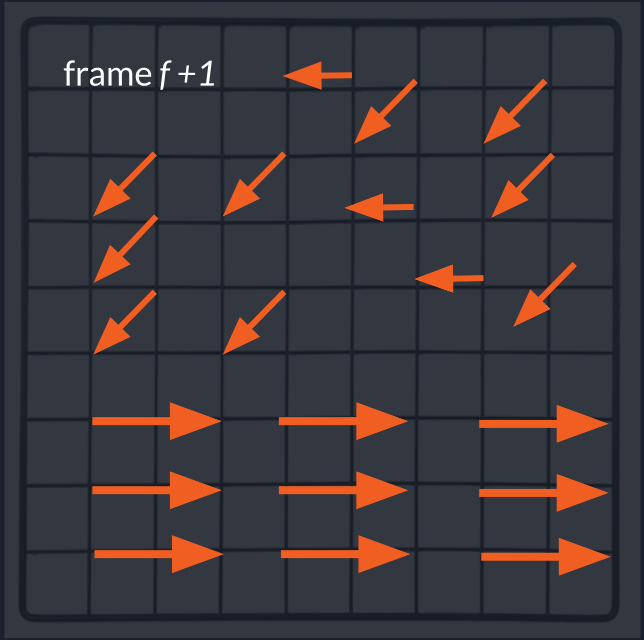
- Motion
- Frequency
- Frame saliency

Motion

$$\phi(v_j, f) = OpticalFlow(v_j(f-1), v_j(f))$$

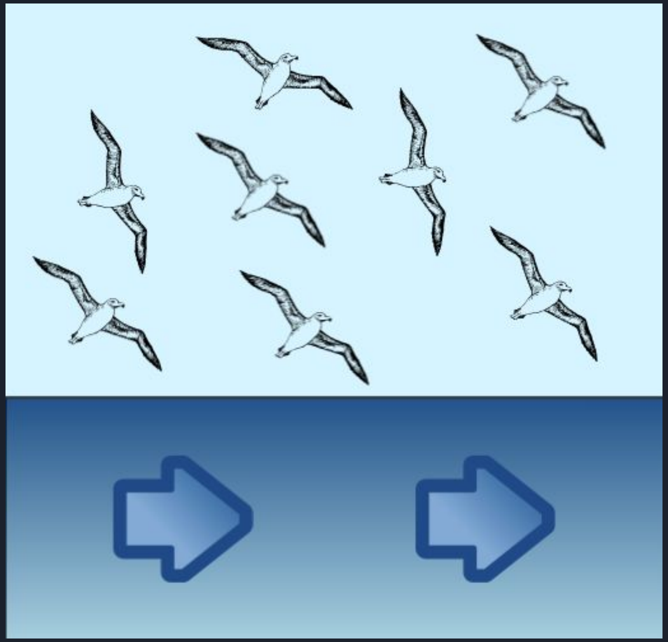
Can we tell from a single frame if it has salient motion?

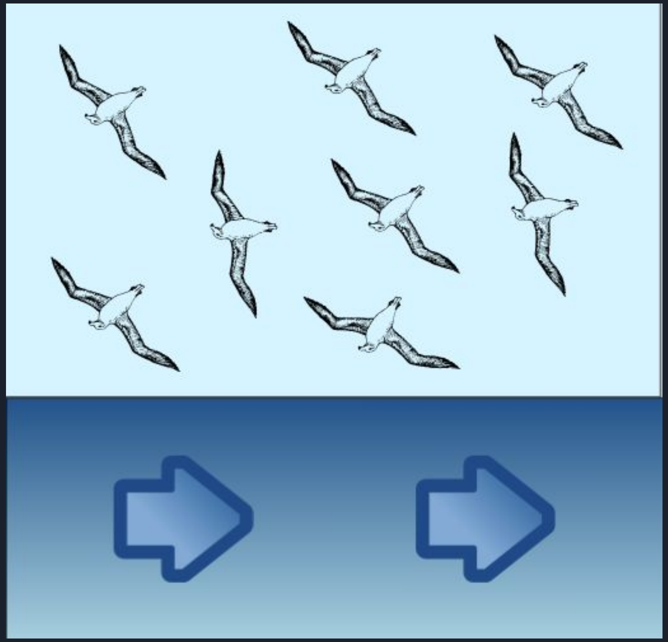




Motion

What is actually the most interesting motion?



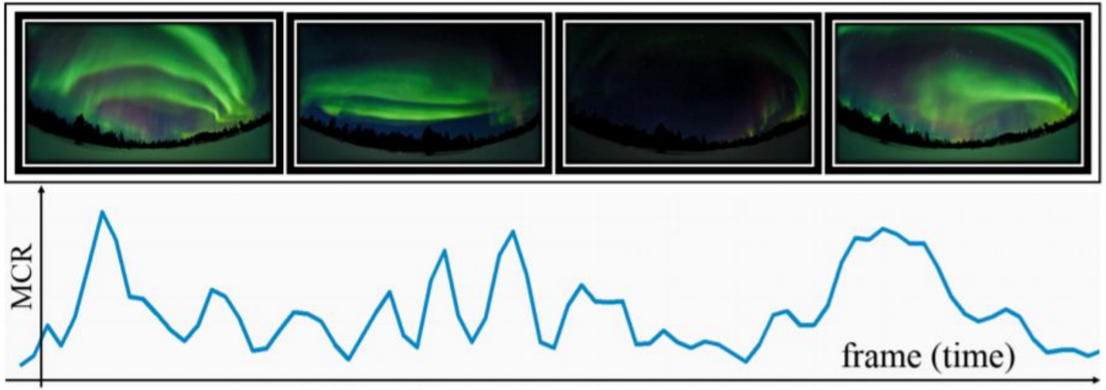


Motion

- What is the difference between the Optical Flow and Motion Change Rate (MCR) ?

(weighted mean)

$$\begin{aligned} \nabla\phi(v_j, f; x) &= \phi(v_j, f; x) - \phi(v_j, f-1; x'), \qquad (2) \\ \text{where} \quad x &= x' + \phi(v_j, f-1; x'). \end{aligned}$$



Motion - MCR

$$\begin{aligned} \nabla\phi(v_j, f; x) &= \phi(v_j, f; x) - \phi(v_j, f-1; x'), \qquad (2) \\ \text{where} \quad x &= x' + \phi(v_j, f-1; x'). \end{aligned}$$

pixelwise temporal difference of the optical flow = 

Optical flow

[Real time optical flow with Video++ @ 200 fps]

Mean saliency weighted motion change

a scalar value for the MCR

$$\Phi(v_j, f) = \frac{1}{N} \sum_{x,y} \alpha(v_j, f; x, y) \|\nabla\phi(i, f; x, y)\| / max_f \Phi(v_j, f)$$

saliency map as a weight

what is happening here?

Saliency map

What is a saliency map?

- Represents what is meaningful in the frames

- Using the method in [Cheng et al. 2014)

[Saliency Mapping of Taylor Swift's 'Shake It Off']

Usage of Optical Flow

What else can we calculate once we have the optical flow?

From the optical flow:

- calculate **Motion Change Rate** (MCR)
  - **peak frequency**
- determine **flow peak**
- calculate **dynamism**

Flow Peak & Dynamism

Flow Peak:

$$\Phi(v_j, f) \; = \tfrac{1}{N}\sum_{x,y} \alpha(v_j, f; x, y) \|\nabla\phi(i, f; x, y)\| / max_f \Phi(v_j, f)$$

Dynamism:

$$\varphi(v_j, f) \; = \; prctile(\alpha(v_j, f)\|\phi(v_j, f)\|, 99.9) / max_f \varphi(v_j, f)$$

Music Analysis 3 steps

(1) divide the music piece into several segments

For each segment:

(2) Determine saliency score
(3) Compute features (for defining the transition cost)

Music Analysis - Segmentation

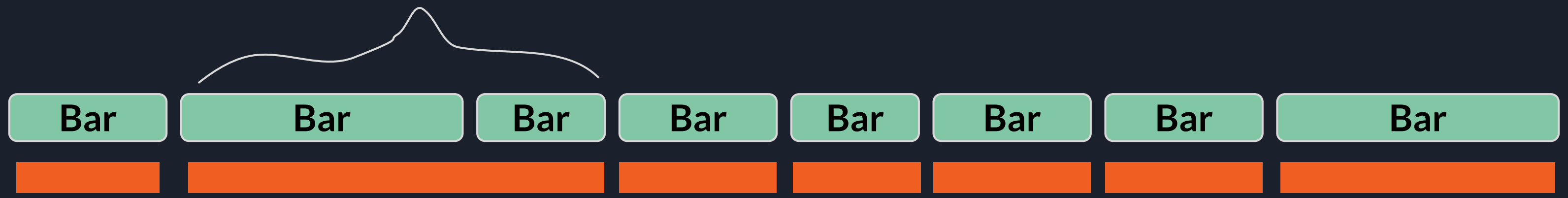
Hierarchical clustering tree:

- Merge the pair of consecutive **segments** with the minimum segment distance

Music Analysis - Segmentation

Hierarchical clustering tree:

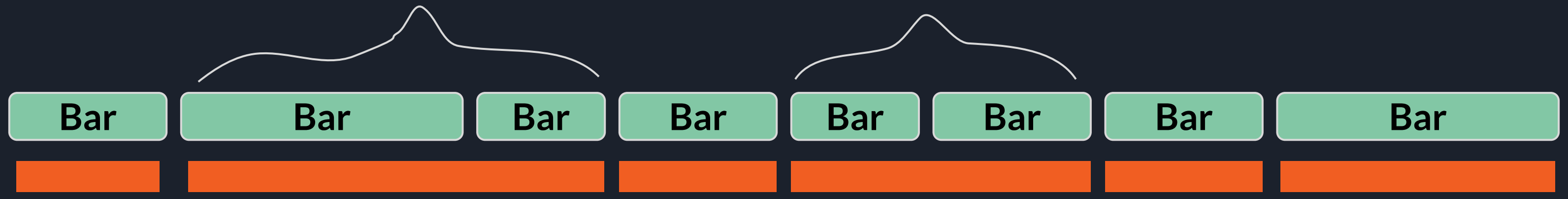
- Merge the pair of consecutive segments with the minimum segment distance



Music Analysis - Segmentation

Hierarchical clustering tree:

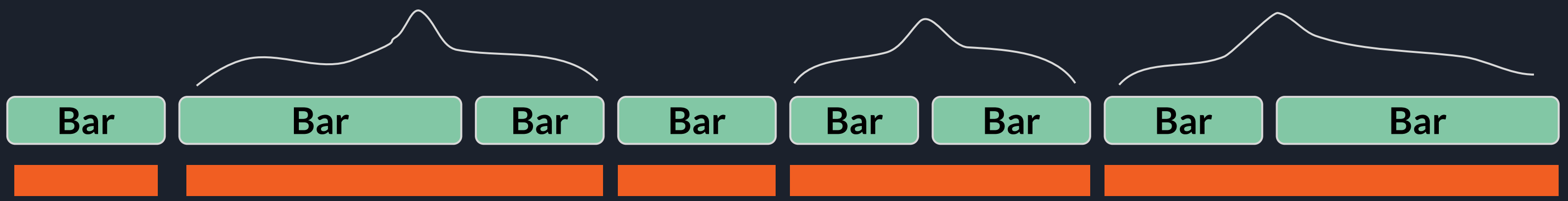
- Merge the pair of consecutive segments with the minimum segment distance



Music Analysis - Segmentation

Hierarchical clustering tree:

- Merge the pair of consecutive segments with the minimum segment distance



Music Analysis - Segmentation

Hierarchical clustering tree:

- Merge the pair of consecutive segments with the minimum segment distance

Music Analysis - Segmentation

Hierarchical clustering tree:

- Merge the pair of consecutive segments with the minimum segment distance

(let's say we are happy with 3 segments)

Segment distance definition:

$$\chi(m_i, m_{i+1}) = w_0 \frac{|\text{pace}(m_i) - \text{pace}(m_{i+1})|}{\text{mode}(pace)} + w_1 \frac{|\text{median}(\text{pitch}(m_i)) - \text{median}(\text{pitch}(m_{i+1}))|}{\sigma_{pitch}} + w_2 \frac{|\sigma(\text{pitch}(m_i)) - \sigma(\text{pitch}(m_{i+1}))|}{\sigma_{pitch}}, \quad (5)$$

Music Analysis - Saliency scores

Eight types of binary saliency scores for **note onsets.**

Initially set to zero

0	0	0	0	0	0	0	0
score_1	score_2	..					score_8

Saliency scores

		if	
pitch-peak	0	..highest pitch > 2x highest pitch at preceding/following note	1
before-a-long-interval	0	..**following** note onset is at least one beat away	1
after-a-long-interval	0	..**preceding** note onset is at least one beat away	1
start-of-a-bar	0	..it is the **first** note onset within a bar.	1
start-of-a-new-bar	0	..it is the first note onset within a NEW bar.	1
start-of-a-different-bar	0	..it is the first note onset within a bar with a **different pattern**	1
pitch-shift	0	..consecutive bars match & more than 90% positions maintain	1?
deviated-pitch	0	..consecutive bars match & pitch difference > σ	1

Music Analysis - Final saliency score

Final saliency score for note onset ti

$$\omega(t_i) = (1 + \mathrm{vol}(t_i) \sum_{i=1}^{8} \mathrm{score}_i)/max(\omega(t_i)), \qquad (6)$$

vol(·) = volume of note = mean squared magnitude in the first 20% of the note duration

Music Analysis - Final saliency score 2.0

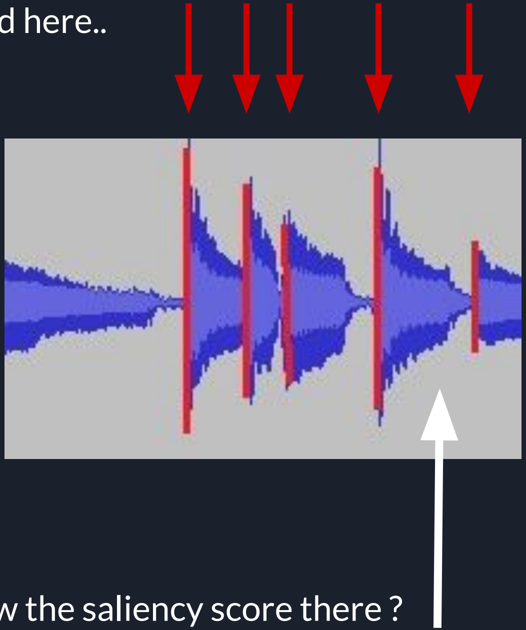
We already have the “final saliency score” - so what is happening here?

$$\Omega(m_i;t) = \sum_{t_i=1}^{K} \omega(t_i) G(t - t_i; \sigma_{t_i}), \qquad (7)$$

G = Gaussian kernel with σti as the standard deviation, centered at time ti

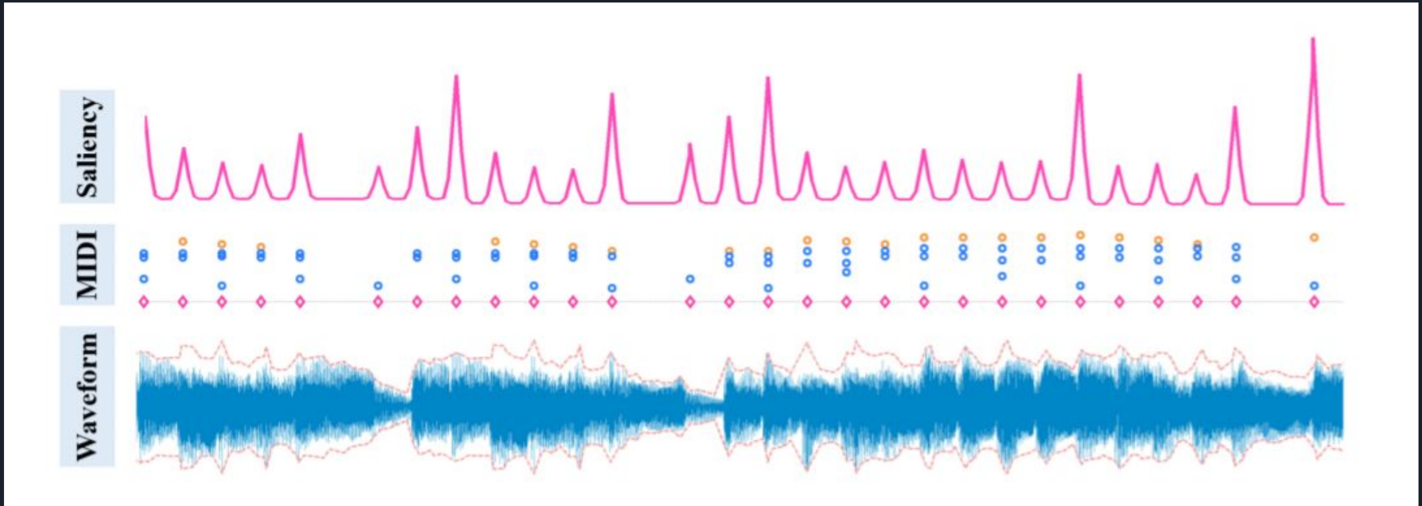
Music Analysis - Final saliency score 2.0

Saliency scores are calculated here..



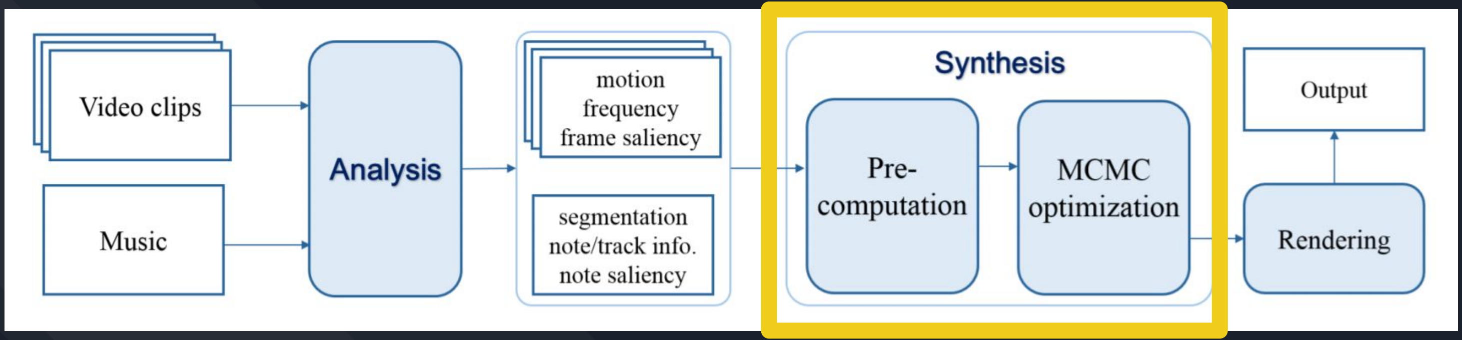
.. But what if we want to know the saliency score there ?

Computed saliency with its associated waveform data



- Could you interpret the saliency by just looking at the waveform, as the manually cut-to-the-beat approach?

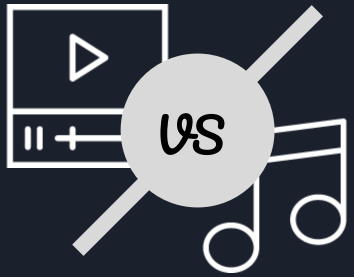
Synthesis


Video clips
Music
Analysis
motion
frequency
frame saliency
segmentation
note/track info.
note saliency
Synthesis
Pre-
computation
MCMC
optimization
Output
Rendering

Recall - energy function to minimize:

$$E(\boldsymbol{\theta}, \mathrm{M}) = E_{match}(\boldsymbol{\theta}, \mathrm{M}) + E_{transit}(\boldsymbol{\theta}, \mathrm{M}) + E_{global}(\boldsymbol{\theta}, \mathrm{M}), \tag{1}$$

Matching cost



What is the purpose of the matching cost?

- We want the“ups and downs” of a video sequence strongly correlate with those of the corresponding music segment.

- peak frequency (video)
- motion change rate (video)

- pace (music)
- saliency score (music)

Energy terms - Matching cost

$$E_{match}(\boldsymbol{\theta}, \mathbf{M}) = \sum_{i=1}^{q} \text{Match}(m_i, \theta_i),$$

$$\Gamma(m_i, \theta_i) = \begin{cases} 1, & pace(m_i) > 1, \psi(\theta_i) < 0.5; \\ 1, & pace(m_i) < -1, \psi(\theta_i) > 2; \\ 0, & \text{otherwise}, \end{cases}$$

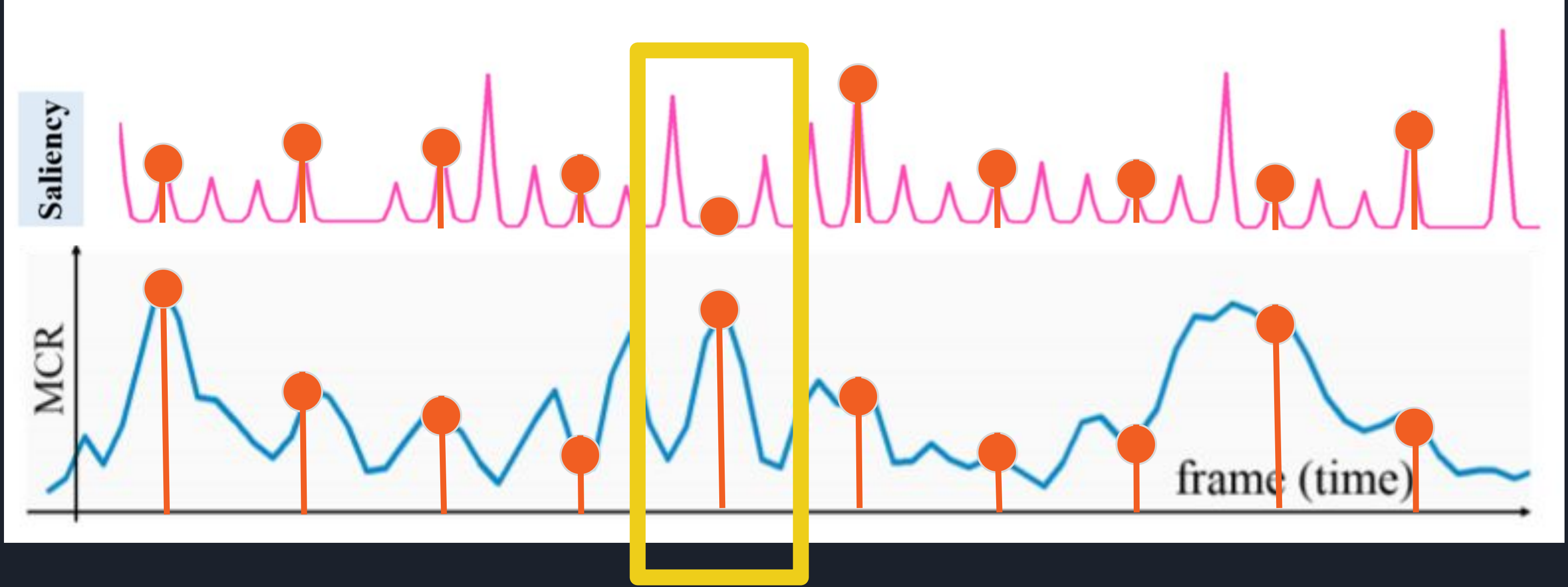
$$\text{Match}(m_i, \theta_i) = \Psi(m_i, \theta_i) + \Gamma(m_i, \theta_i).$$

$$\Psi(m_i, \theta_i) = \begin{cases} G(x; \sigma_{co}), & x \geq 0; \\ 2 - G(x; \sigma_{co}), & x < 0; \end{cases}$$

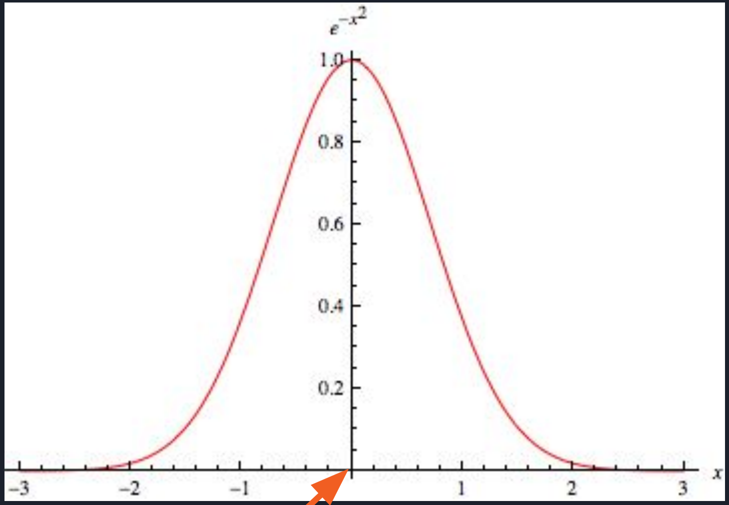
$$x = \Omega(m_i, t)^T \Phi(a_i, t) / N$$

Saliency/MCR mismatch

$$x = \Omega(m_i, t)^T \Phi(a_i, t)/N$$

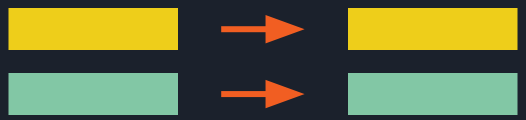


Saliency/MCR mismatch



.. and if x = 0 we will get maximum penalty cost from the Gaussian kernel

Transition cost



What is the purpose of the transition cost?

- We want to encourage video transitions across cuts to match characteristics of musical transitions across segments

- “velocity” = mean flow magnitude (video)
- dynamism (video)

- pace (music)
- number of tracks (music)

Energy terms - Transition Cost

$$\text{Transit}(i, i+1) = \Delta(m_i, m_{i+1}, \theta_i, \theta_{i+1}) + \Lambda(m_i, m_{i+1}, \theta_i, \theta_{i+1}).$$

$$\Delta(m_i, m_{i+1}, \theta_i, \theta_{i+1}) = \begin{cases} 1, & \kappa_p < 0.5, \kappa_v > 0.75; \\ 1, & \kappa_p > 2, \kappa_v < 1.5; \\ 0, & \text{otherwise,} \end{cases}$$

$$\Lambda(m_i, m_{i+1}, \theta_i, \theta_{i+1}) = \begin{cases} 1, & \nabla t < 0, \nabla d > -0.3; \\ 1, & \nabla t > 0, \nabla d < 0.3; \\ 0, & \text{otherwise,} \end{cases}$$

$$\kappa_p = \text{pace}(m_{i+1}) / \text{pace}(m_i)$$

$$\kappa_v = \text{vel}(p_{i+1}) / \text{vel}(p_i)$$

$$\nabla t = \text{numtrack}(m_{i+1}) - \text{numtrack}(m_i)$$

$$\nabla d = \delta(v_{a_{i+1}}, sf_{i+1}) - \delta(v_{a_i}, ef_i)$$

Global constraints

What is important to achieve an interesting composition?

- using the same video clips over and over again while ignoring others is probably not desirable ..

Introducing a penalty cost to prevent duplicates:

$$\sum_{i=1,..,p} \left(2^{\text{count}(v_i)-1} - 1\right)$$

Optimization

Recall - what to optimize

Once again, what has to be optimized?

These parameters! $\theta_i \;=\; (v_{a_i}; sf_i, ef_i, scale_i)$

packed with a lot of features now

Too large parameter space for the Metropolis-Hasting algorithm to traverse
-> Introduce a precomputation step:

For each possible music-video pair, the optimal 4-tuple of these parameters is computed

Optimization - precomputation step

For each music-video candidate pair:

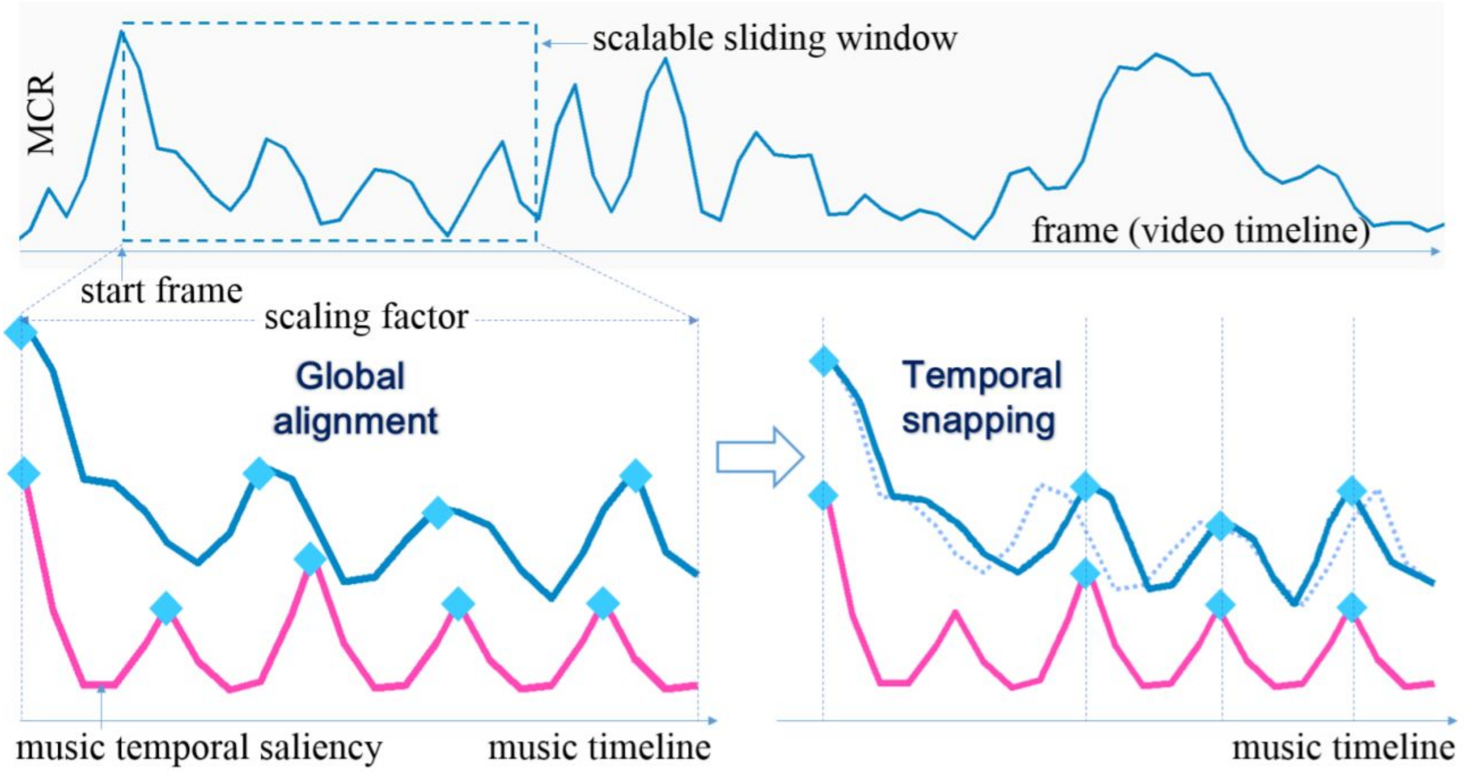
Global alignment

- Optimizes the position of the first frame and its associated temporal scaling factor

Temporal snapping

- With the global alignment result, now allow a temporally varying scaling factor for better synchronization

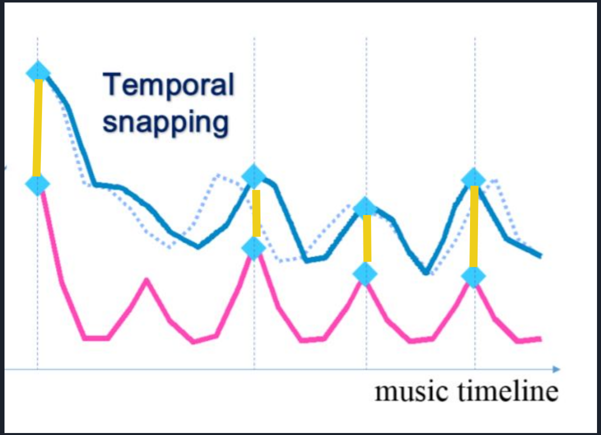
Global alignment & Snapping



Temporal Snapping

Identifies a set of keyframes in the video and optimizes a temporal scaling between them to match note onsets.

The following video frames are chosen as keyframes, (a1) the starting frame, (a2) the last frame, and (a3) any intermediate frame whose motion change rate is a local peak, i.e. large than that of the preceding and following frames and above the 90 percentile. Likewise, the following note onsets are labeled as salient, (b1) the first one of a music segment, (b2) the last one of a music segment, and (b3) any note onset with a saliency score 0.5 or above.



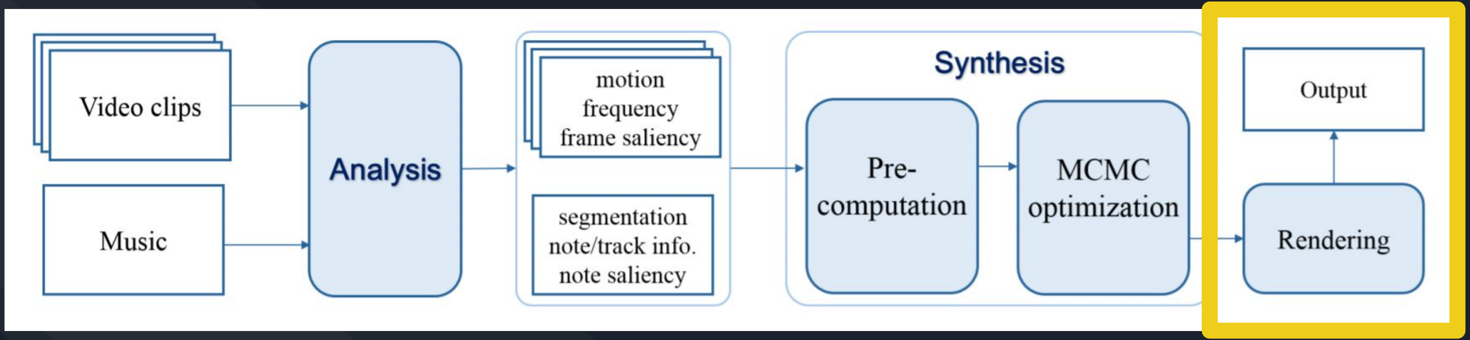
MCMC sampling

Final step is to sample the label space for an optimal solution

Two types of mutations are design:

- with probability 0.7, the video index for a music segment is updated to a random index between 1 and n, where n is the total number of video clips
- and with probability 0.3, two music segments' corresponding video indices are swapped.

Rendering



Video clips
Music
Analysis
motion
frequency
frame saliency
segmentation
note/track info.
note saliency
Synthesis
Pre-
computation
MCMC
optimization
Output
Rendering

Rendering

The final video montage is formed by concatenating the scaled subsequences

- Given θ and the temporal snapping parameters, upsampling and downsampling are applied

[Icon made by Freepik from www.flaticon.com

Results

Results

Recall Visual Rhythm and Beat (Davis et al.)

Commonalities ?

Differences ?

How important is video rhythmic in the two implementations?

- what kind of inputs are expected for the two applications?

Finito!