Where is the Drop: Segment Boundary Detection in Electronic Dance Music

Final Project Report for MUMT 621 Music Information Retrieval

Zeyu Li

Abstract

In Electronic Dance Music (EDM), Disc Jockeys (DJs) often rely on segment boundaries to mix music for recordings and live shows. In this project, we apply a novelty-based approach to accurately detect the segment boundaries in EDM, exploiting temporal and structural elements of this music genre. Novelty curves are computed using a combination of feature sets, autocorrelation, self-similarity matrices, and kernels. For tempo estimation and beat tracking, a novelty curve with a short hop size is applied. Novelty curves of beat-level audio features are computed for downbeat detection and boundary detection. The detected boundaries are then adjusted to match the downbeat onsets. In-house and third-party datasets are tested to evaluate this model, and each step of the process is evaluated separately. We show that this process performs well under small temporal tolerance level.

1. Introduction

Music segmentation is the process of segmenting a full music piece into parts based on the structural similarity within each part, as segment boundaries often occur when the music pattern changes. While humans tend to easily discover patterns and find structures in the information they perceive, it is not as straightforward to emulate this process using computer algorithms. Many techniques (Jouni et al. 2016) are proposed for the task of automatic music segmentation, although they often aim to provide fuzzy boundaries for multiple different genres, therefore would fall short when accurate segment boundaries are required.

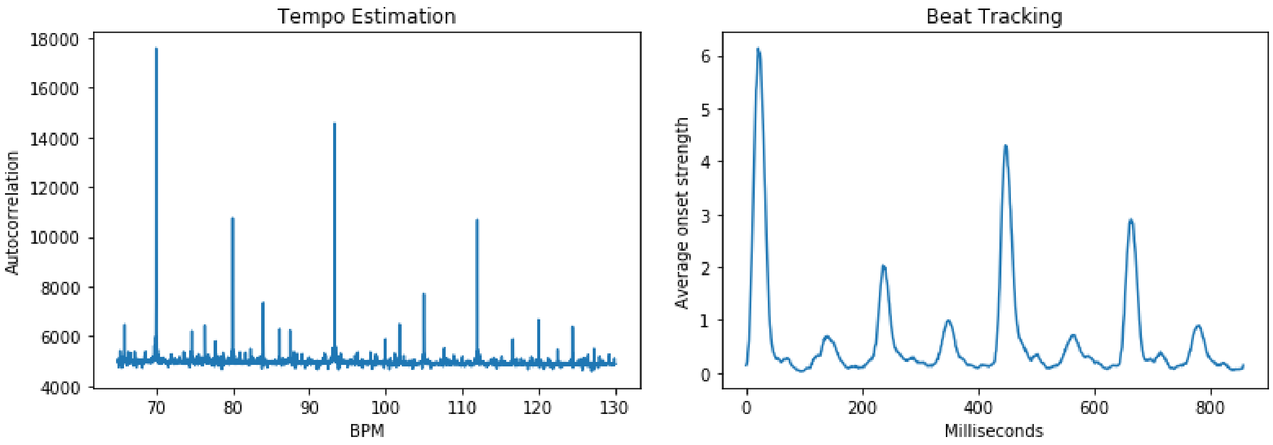
In Electronic Dance Music (EDM), as most tracks are made to be mixed with others for a continuous dancing experience, their music structure is often made simple, and shares common properties such as using a constant tempo, and 4/4 meter throughout a track (Butler 2006, 113–116). As a result, Disc Jockeys (DJs) often rely on structural segmentation to mix EDM tracks smoothly, by creatively connecting and overlapping different segments throughout the course of a performance (Butler 2006, 197). These segment boundaries often occur when the song's perceived energy changes, and they are known as buildup (energy increase), drop (energy peak), and breakdown (energy decrease) among others (Butler 2006, 222).

Rocha et al. (2013), and Vande Veire (2017) exploited temporal and structural elements of the EDM genre by breaking down the segmentation problem into multiple stages, such as beat tracking, downbeat detection, and segmentation. Yadati et al. (2014) applied an SVM classifier to find the onset of the drops. Yet these studies still allowed large temporal tolerances of at least 0.5 seconds (s), which would be an audible discrepancy if these results are applied to beat matching in DJ mixing (Butler 2006, 56). To resolve this problem, we follow the multi-stage process and apply a novelty-based approach to each task, and evaluate its performance.

2. Tempo Estimation

We assume all EDM follows a constant tempo and metre from start to finish, we can therefore discover the tempo through correlating the audio signal with the delayed version of itself. We follow the approach introduced by Davies and Plumbley (2007) and revised by Vande Veire (2017) in 4 steps:

a. Compute the onset detection function of each audio frame at a hop size of 512 samples, based on Mel-spectrum with a maximum frequency of 400 Hz in 8 bins
b. Compute the adaptive mean of the onset strength with a window size of 16 frames, and set all frames with onset strength below the mean to zero. This creates an onset novelty function
c. Compute the frame-level autocorrelation of the novelty function at a range of hypothetical tempi, averaged by the number of repetitions for each given tempo, and we get a collection of autocorrelation value of each tempo, shown in Figure 1 (a)
d. The tempo with the highest peak in the collection is chosen as the detected tempo



(a) **(b)**

Figure 1. (a) Autocorrelation at each hypothetical tempo and (b) average onset strength at each millisecond

Experiments show that although this approach generally yields accurate tempo, in a few tracks the tempo at 3/2 or 3/4 of the annotated tempo has a higher autocorrelation value, which is a result of off-beat onsets in the music. In this case we pick the second highest peak as the tempo if it is 2/3 or 4/3 of the tempo with the highest peak, and its autocorrelation value is more than 80% of the highest peak.

3. Beat Tracking

To complete beat tracking, the next step after tempo estimation is to determine the onset when each beat starts. Since the beats are evenly spaced given a constant tempo, this would be a trivial task once we know when the first beat starts. To do that, we sum up our frame-level novelty function at evenly spaced beat intervals starting with different delays, and the result is a collection of "onset strength" for each different delay, as shown in Figure 1 (b). Then the highest peak of this collection is chosen as the onset of the first beat.

Tests show that our method occasionally yields the location of off-beat, instead of on-beat. To fix this issue, we observe that the first beat often starts right after an EDM song starts (with no delay), therefore we take the peak within the first 1/4 of beat interval as the start of the first beat, if its strength is more than 80% of the highest peak.

4. Downbeat Detection

In EDM, DJs generally marks the segment boundaries on the downbeat (the first beat in a measure of 4 beats), as this makes it easy to match and mix EDM tracks. Therefore, we assume that segment boundaries all occur on the downbeat, and that the downbeat occurs at every 4 beats throughout the track. Instead of using a classifier-based approach, such as those introduced by Hockman et al. (2012) and Böck et al. (2016), we apply a novelty-based approach to find the downbeats in a similar way as we did for beat tracking.

With the results of tempo estimation and beat tracking, we can now compute a new onset novelty curve based on the features of each beat. This is done through first computing onset value with a feature set at the equally spaced beat intervals, and taking the difference in onset value of each beat to its previous one as the novelty value. Tests have shown that the best feature set for this task is the Constant-Q transform (CQT), with a minimal frequency of 30 Hz, and 20 frequency bands (20/12 octaves).

Once we have the beat-level novelty curve, we take the top 5 peaks from it, and determine which one out of the 4 beats in a measure the peaks belong to. The beat with the most peaks is chosen as the downbeat, unless when there are two beats each with 2 peaks, then the first one of the two beats is chosen as the downbeat. The downbeat detection is shown in Figure 2.

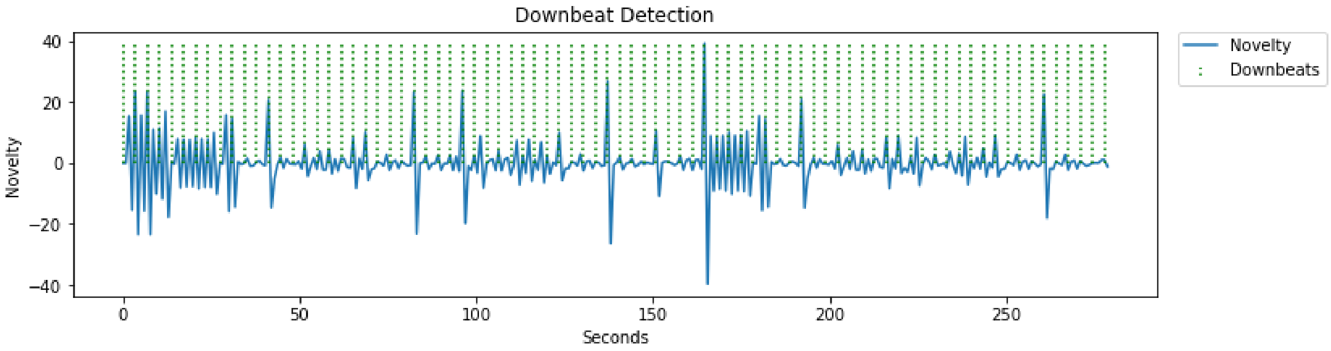


Figure 2. Beat-level novelty function and downbeat locations

5. Boundary Detection

Next, we finally address the problem of music segmentation. This subject has been studied extensively, and the proposed approaches generally belong to 3 categories: novelty-based approaches such as Foote and Cooper (2003), Rocha et al. (2013), and Scarfe et al. (2013, 2014), state-based approaches such as Ullrich et al. (2014), and repetition-based approaches such as Todd (1994) and Goto (2003). A more extensive review can be found by Jouni et al. (2016).

For our task of EDM segmentation, we use a novelty-based approach that is first proposed by Foote and Cooper (2003), and used by Rocha (2013), Scarfe (2013, 2014), and Vande Veire (2017). Using results of previous steps, we first compute a feature vector at each beat, and construct a self-similarity matrix (SSM) of the feature vector using cosine distance function. A novelty curve is then constructed by convoluting the SSM along its diagonal with a checkerboard kernel. The peaks of the novelty curve are chosen as the raw segment boundaries. We found the best feature set for segmentation is MFCC (using all bands), and the best checkerboard kernel size is 32 beats. The detected raw boundaries are shown in Figure 3.

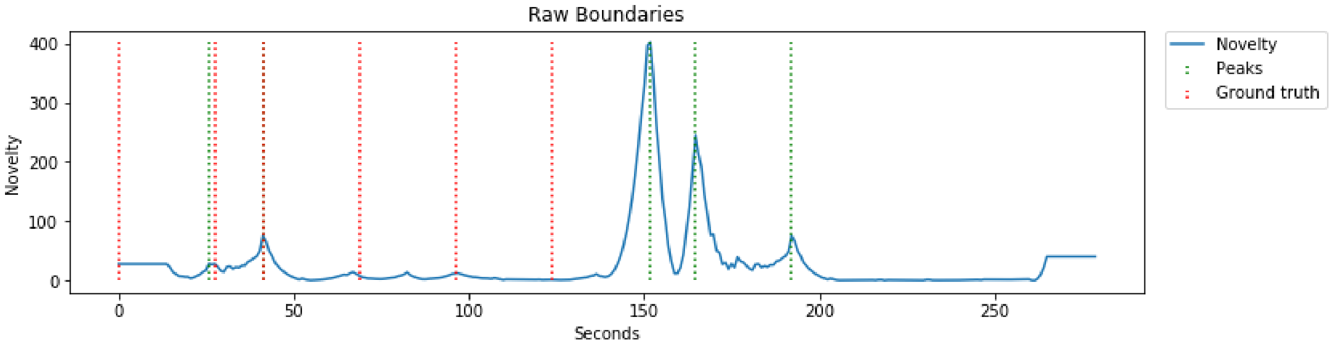


Figure 3. Novelty function for segmentation and detected raw boundaries

6. Boundary Adjustments

As the feature sets and novelty curves are computed at beat level, and our assumption is that the boundaries only occur on the downbeat, we can therefore adjust the boundaries to the nearest downbeats. When the identified boundary is at the middle of the measure, it is shifted to the downbeat located before itself. After the adjustments, we also add a boundary between 2 consecutive boundaries every 8 measures, if the distance between them is more than 12 measures. The adjusted segmentation is shown in Figure 4.

Rocha (2013) and Vande Veire (2017) also proposed that the detected raw boundaries to be adjusted based on musically informed rules as the final step of the segmentation process. These rules include assuming that major segments have lengths of multiples of 4 or 8 measures, etc. However, experiments show that the adjusted boundaries are often even further away from the ground truth, because while the newly adjusted boundaries do follow these rules, they are often adjusted to the opposite direction, therefore this step is discarded.

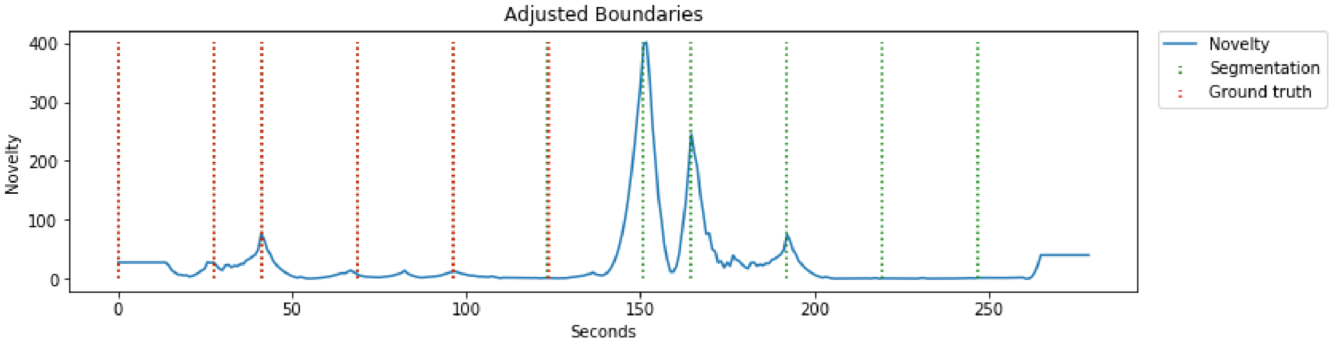


Figure 4. Adjusted segmentation and detected raw boundary peaks

7. Evaluation

A few datasets are employed for the evaluation of this project. The author has an in-house dataset of 214 popular EDM from multiple live performances. Genre-wise, there are 115 Tropical House tracks, 50 Trap, and 49 Dubstep, and the number of segment boundaries for each genre is 546, 236, and 282, respectively. However, not all "ground truth" segment boundaries are annotated, instead only less than half of them are marked for live performance. The annotation is imbedded in the MP3 audio files using the Serato DJ[1] software, and extracted using eyeD3[2].

We also obtained the dataset used by Rocha et al. (2013), which is comprised of 35 popular EDM songs of various different EDM genres. We found the audio files of 32 annotated songs online, and determined that they match the original songs used in the dataset. This give us a total of 379 annotated boundaries. The annotation is provided in the SDIF format, and is extracted through OpenMusic[3].

In Table 1, we report the accuracy of the tempo estimation, beat tracking, downbeat tracking, and segmentation separately. For tempo estimation, we use a range of tempo from 65 to 129.99 beats per minute (BPM) at increment of 0.01 BPM, and allow a tolerance level of 0.1 BPM. We also assume that any tempo reported as twice or half of the annotated tempo as success, and this allows us to report the tempo in almost any range. For beat tracking and downbeat detection, we use a temporal tolerance of 0.1s and 0.5s respectively. In this case, it should be noted that even when downbeat detection succeeds, beat tracking could still fail as a result of using a smaller tolerance. For segmentation, we report the recall rate, precision rate and the F-score for tolerances of 3.0s, 0.5s, and 60 milliseconds (ms). Precision and F-score are not reported for the in-house datasets as the annotation is not complete. Results of 60ms tolerance for the dataset from Rocha et al. (2013) are not reported due to the quality of the dataset.

As can be seen, we have successfully estimated the correct tempo for 100% of songs in our test datasets given a small tolerance level. Beat tracking and downbeat detection also perform well, and the results are close to state-of-the-art methods developed for multiple music genres (Böck et al. 2016). For segmentation, although our results for the Rocha et al. (2013) dataset do not match up to their original

[1] Serato DJ: https://serato.com/dj
[2] eyeD3: https://eyed3.readthedocs.io/en/latest/
[3] OpenMusic: http://repmus.ircam.fr/openmusic/home

results, we argue that this is because we developed different beat tracking and downbeat detection methods, and we can see our method is more suitable for our in-house datasets, and the reported recall rates are close to the original method. We also show that the loss of accuracy is small when the segmentation tolerance is set to 60ms, as the results are similar to those when the tolerance is 500ms.

	Tropical House	Trap	Dubstep	Rocha et al.
Number of Songs	115	50	49	32
Number of Segment Boundaries	546	236	282	379
Tempo Estimation	100.00%	100.00%	100.00%	100.00%
Beat Tracking	92.17%	84.00%	100.00%	78.13%
Downbeat Detection	80.00%	82.00%	83.67%	84.38%
Segmentation – Recall (±3.0s)	84.07%	88.14%	59.22%	71.77%
Segmentation – Precision (±3.0s)	N/A	N/A	N/A	58.87%
Segmentation – F-Score (±3.0s)	N/A	N/A	N/A	64.68%
Segmentation – Recall (±0.5s)	57.51%	59.75%	45.39%	36.94%
Segmentation – Precision (±0.5s)	N/A	N/A	N/A	30.30%
Segmentation – F-Score (±0.5s)	N/A	N/A	N/A	33.29%
Segmentation – Recall (±60ms)	52.75%	53.81%	45.39%	N/A

Table 1

8. Conclusion

Through novelty-based methods and taking advantage of the music structure of EDM, we have developed a process for tempo estimation, beat tracking, downbeat detection, and segmentation of this popular music genre. Although the method is developed mainly for EDM, it can be readily applied to other genres of music with a constant tempo.

The main contribution of this project is we developed a segmentation method that works for small temporal tolerance levels as well as larger ones, and it achieves similar results to state-of-the-art segmentation results previously published. The low-tolerance result is important because it is required when it comes to beat matching for music mixing, and the tolerance level allowed by existing literature on music segmentation is generally set too large.

Like many other music information retrieval applications, this process does not take into consideration individual perceptual differences, therefore it is not suitable for applications when such differences are considered significant. As noted by Casey et al. (2008) and Schedl (2013), user-focused music information retrieval techniques would contribute to applications that better serve them, even though this is an area still largely unexplored in music segmentation.

Bibliography

Böck, Sebastian, Florian Krebs, and Gerhard Widmer. 2016. Joint beat and downbeat tracking with recurrent neural networks. In *Proceedings of the International Society for Music Information Retrieval Conference.*

> State-of-the-art algorithm using RNN for downbeat tracking based on low-level features. Tested on a large dataset of songs from different genres.

Butler, Mark J. 2006. *Unlocking the groove: Rhythm, meter, and musical design in electronic dance music.* Bloomington, IN: Indiana University Press.

> Comprehensive study of EDM musical form from a theoretical perspective.

Casey, Michael A., Remco Veltkamp, Masataka Goto, Marc Leman, Christophe Rhodes, and Malcolm Slaney. 2008. Content-based music information retrieval: Current directions and future challenges. *Proceedings of the IEEE* 96 (4): 668–96.

> Review early research on content-based music information retrieval techniques. Comprehensively review MIR-related topics including its use cases, feature representation, audio analysis approaches, and applications.

Davies, Matthew E. P., and Mark D. Plumbley. 2007. Context-dependent beat tracking of musical audio. *IEEE Transactions on Audio, Speech, and Language Processing* 15 (3): 1009–20.

> Extract the beat locations based on an onset detection curve directly computed from the audio. The algorithm can be applied to any other onset curves.

Foote, Jonathan T., and Matthew L. Cooper. 2003. Media segmentation using self-similarity decomposition. In *Proceedings of the SPIE Conference on Storage and Retrieval for Multimedia Databases.*

> Introduce Self-Similarity Matrix, checkerboard kernel, and novelty function to study music and media segmentation.

Goto, Masataka. 2003. A chorus-section detecting method for musical audio signals. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*.

> Introduce a popular approach called RefraiD that enhances the similarity stripes on a time-lag matrix to detect music chorus sections. Test on the RWC Popular Music database shows 80% accuracy.

Hockman, Jason A., Matthew E. P. Davies, and Ichiro Fujinaga. 2012. One in the jungle: Downbeat detection in hardcore, jungle, and drum and bass. In *Proceedings of the International Society for Music Information Retrieval Conference.*

> A genre-specific downbeat detection program using PCA-selected high-level features and SVM. Tested using a dataset annotated by a professional drum and bass musician.

Jouni, Paulus, Meinard Müller, and Anssi Klapuri. 2016. Audio-based music structure analysis. In *Proceedings of the International Society for Music Information Retrieval Conference.*

> Review research on audio-based music structure analysis. Summarize common feature representation and SSM. Categorize approaches into novelty-based, repetition-based and homogeneity-based. Evaluate on MIREX dataset.

Rocha, Bruno, Niels Bogaards, and Aline Honingh. 2013. Segmentation and timbre similarity in electronic dance music. In *Proceedings of the International Sound and Music Computing Conference*.

> Extract EDM segmentation via beat/downbeat detection, beat-level novelty detection, and segment boundary adjustments based on downbeat locations. Evaluated using in-house, RWC and Eurovision datasets.

Scarfe, Tim, Wouter M. Koolen, and Yuri Kalnishkan. 2013. A long-range self-similarity approach to segmenting DJ mixed music streams. *Artificial Intelligence Applications and Innovations IFIP Advances in Information and Communication Technology* 412: 235–44.

Scarfe, Tim, Wouter M. Koolen, and Yuri Kalnishkan. 2014. Segmentation of electronic dance music. *International Journal of Engineering Intelligent Systems for Electrical Engineering and Communications* 22 (3/4): 1–18.

> The above 2 papers apply similarity matrix and novelty curve to segmentation of DJ mixes. Evaluated on freely available dataset with crowd-sourced annotation and DJ's own annotation.

Schedl, Markus, Arthur Flexer, and Julián Urbano. 2013. The neglected user in music information retrieval research. *Journal of Intelligent Information Systems* 41 (3): 523–39.

> Review current literature (and lack thereof) on user-centric MIR, particularly similarity analysis. Call for more work in a few directions: user models, personalization, multifaceted similarity measures, and evaluation.

Todd, Neil P. M. 1994. The auditory "Primal Sketch": A multiscale model of rhythmic grouping. *Journal of New Music Research* 23 (1): 25–70.

> Introduce rhythmogram to represent the grouping of a sequence of events. Also apply it to study music perception through a multiscale decomposition of the auditory nerve response.

Ullrich, Karen, Jan Schlüter, and Thomas Grill. 2014. Boundary detection in music structure analysis using convolutional neural networks. In *Proceedings of the International Society for Music Information Retrieval Conference.*

> Apply CNN to discover music boundaries on the SALAMI dataset, which is a large and diverse collection of songs of different genres. The accuracy of the segmentation results is below 65%, even though it outperforms other existing methods.

Vande Veire, Len. 2017. From raw audio to a seamless mix: An artificial intelligence approach to creating an automated DJ system. Master’s thesis, Ghent University.

> Create an automatic DJ system through beat/downbeat tracking, segmentation, and preprogrammed transition sequences, focusing on drum-and-bass music. Evaluated the crossfading performance through both machine learning and user listening experiments, yet results are inconclusive.

Yadati, Karthik, Martha Larson, Cynthia C. S. Liem, and Alan Hanjalic. 2014. Detecting drops in electronic dance music: Content based approaches to a socially significant music event. In *Proceedings of the International Society for Music Information Retrieval Conference*.

> Detect drops in EDM by applying first unsupervised learning to extract segments, then SVM on MFCC to classify segments into drop vs. non-drop. Tested on in-house dataset.