# Music Composition Inspired by Sea Wave Patterns Observed from Beaches

**C. Florintina and E. S. Gopi**

**Abstract**  Nature has always been a constant source of inspiration for mankind. This work proposes to harness what nature has to offer through the sea waves to compose music. In this work, the features of the sea waves as observed from beaches are utilized to generate musical semitones, without the influence of any other human-composed music. The musical strains so generated are hence completely unique and could serve as a significant plug-in for a composer.

**Keywords**  Music composition · Sea wave pattern · LDS · Kernel technique

## 1  Introduction

Composing a unique strain of music has become a highly commendable feat, and composing music which is not entirely predictable being every composers' wish certainly does not alleviate the difficulties faced. It is quite beyond the reach of small-scale companies to compose a unique theme music and small-scale gaming companies to come up with a unique strain of music for each level of their games, due to the cost involved. The above-mentioned causes justify the need for adopting machine intelligence to aid the field of music composition.

It is quite impossible for automatically composed music to possess the finesse of human composition. Muoz et al. [1] tries to alleviate this problem with the aid of unfigured bass technique. Santos et al. [2] gives a brief account of the different evolutionary techniques used for composing music. Engels et al. [3] introduces a tool which could be of use to small-level game designers that automatically generates

C. Florintina · E. S. Gopi (✉)
Department of Electronics and Communication Engineering,
National Institute of Technology Trichy,
Tiruchirapalli 620015, Tamil Nadu, India
e-mail: esgopi@nitt.edu

C. Florintina
e-mail: florintinachaarlas@gmail.com

unique music for each level of game. Roig et al. [4] extracts musical parameters from an existing piece of music and designs a probabilistic model to generate unique music. Merwe et al. [5, 6] use Markov models and [7] uses genetic algorithms to automate music composition. Pearce and Wiggins [8] proposes a framework to evaluate machine-composed music. Florintina and Gopi [9] proposes to design a simple linear discrete system (LDS) as a highly storage and computation-efficient classifier in kernel space.

In the following section, we disclose the methodology used to generate music and the details of the classifier used. The extractable features are presented in Sect. 3. Section 4 talks of the experiments done, and Sect. 5 includes the results and conclusions.

## 2 Methodology

A video of the sea waves as observed from the beach is chosen. The frames are collected with a gap of one second duration (refer Fig. 1). In order to analyze the wave patterns, initially the waves need to be demarcated. To do so, the linear discrete system (LDS) [9] is chosen to function as a classifier in kernel space. To train the classifier, the first frame is taken and random blocks of the desired region (waves) are manually collected into an image to function as training data for desired class. Similarly, training data for the undesired class is also accumulated as in Fig. 2. These training data are used to train the LDS classifier. By feeding the rest of the frames as input into the trained LDS, wave-demarcated output frames are obtained by assigning value '0' to blocks classified as waves and '1' to the other blocks as depicted in Fig. 2. Upon careful observation, the notable and unique features are acquired from the wave patterns. These features are then mapped into the basic musical semitones following a suitable scale, which could be played using any musical instrument, the notes and rests being left to the musician's choice. This technique is not implemented with the illusion that completely musical audio with appropriate periodic structure would be the outcome. As a matter of fact, it just offers to serve as one of the plug-ins in music composition, upon which the composer could build his composition.

## 2.1 LDS as a Supervised Classifier in Kernel Space

As the name suggests, the classifier is designed as a LDS, wherein the output of the classifier is obtained by simply convolving the input vector with the impulse response coefficients (**h**) of the LDS. The training of this classifier lies in obtaining the appropriate **h**, with which it becomes feasible to precisely classify the input data. Particle swarm optimization (PSO) technique is used to acquire **h**.

In order to employ the LDS as a classifier, the following algorithm (as depicted in Fig. 3) is to be followed.

1. If the training data is of the form of a image, it is divided into blocks of appropriate size and is arranged as vectors, $x_{ij}$, where $i = 1, 2, \ldots n$, $j = 1, 2, \ldots m$ and $n$ and $m$ signify the number of training vectors in each class and number of classes.
2. While collecting the training data, it becomes vital that sufficient amount of data belonging to every class is assimilated, so as to efficiently train the classifier.
3. In order to increase the classification accuracy, Gaussian kernel is used to map the input data to higher dimensional space (HDS). The sigma value to be used in the Gaussian kernel is optimized with the aid of PSO, such that the value of excess kurtosis is minimized, homoscedasticity is maximized, and Euclidean distance between the classes is maximized [10]. This is done so that the possibility of the classes overlapping is reduced as they are made into Gaussian distributed classes which possess identical covariance matrices and are as much separated from each other as possible.
4. What vector (target) the LDS should generate when a particular class data is fed is chosen beforehand. Generally, the target vector is segmented into '$m$'
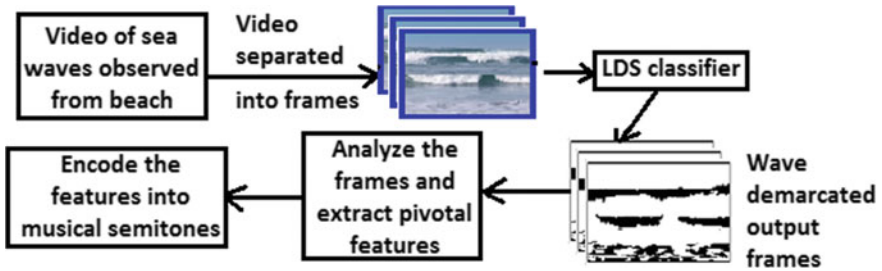


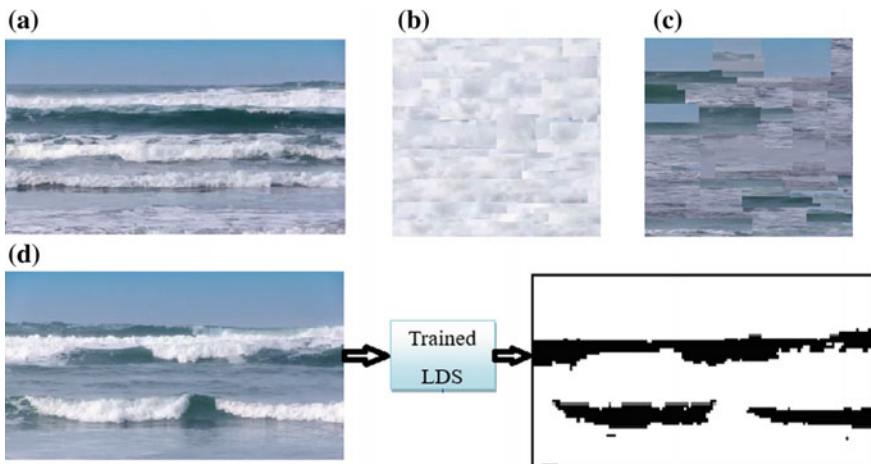**Fig. 1** Methodology adopted to compose music from sea wave patterns



**Fig. 2** **a** First frame. **b** and **c** are the training data for the desired and undesired classes, respectively, collected from (**a**). **d** Demarcation of waves

segments. If $\mathbf{t_j}$ is the target for $j$th class, then it is chosen such that the sum of it's $j$th segment entries is higher than it's other segments' sums as in Fig. 4.

5. The data mapped into the HDS is then given as input to the LDS, and this vector convolved with the LDS's impulse response coefficients ($\mathbf{h}$) gives the output, $\mathbf{y_{ij}}$. Mean square error (MSE) between $\mathbf{y_{ij}}$ and $\mathbf{t_j}$ is calculated; the values of $\mathbf{h}$ are regulated so as to minimize MSE using PSO.
6. The previous step is repeated till MSE becomes as low as possible.
7. The number of coefficients of $\mathbf{h}$ is varied, the training is repeated every time, and finally, the number of impulse response coefficients is dictated by the scenario which gives the maximum percentage of success.
8. The kernel-mapped testing data is then given as input into the trained LDS, and the output is collected.
9. The output is segmented into '$m$' segments and that datum is classified as class $j$, if the sum of the absolute of the $j$th segment's entries is higher than the sum of the absolute of any other segments' entries.

## 3   Extractable Features of Significance

On meticulous analysis of the wave-demarcated frames, it is observed that several spatial and temporal features provide significant information and could be extracted. The spatial features are collected by scrutinizing each and every frame; i.e., the feature extracted from one frame depends neither on the frames preceding it nor
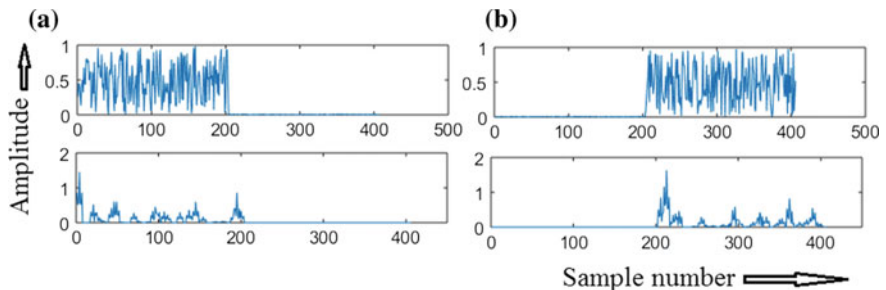


**Fig. 3**  LDS as a classifier in kernel space



**Fig. 4**  First row of **a** and **b** are the intended ones or targets predetermined for class 1 (wave) and 2, respectively. The second row of **a** and **b** is output of the LDS for class 1 and 2 data, respectively, trying to imitate their targets

on the ones following it. Number of waves, thickness of the waves, and distance between the waves observed in each frame are some of the spatial features that could be extracted from the wave patterns. If a feature is collected by analyzing a series of frames, belonging to different times, then it is termed as temporal feature. Time duration for which each wave exists, distance travelled by each wave before it ceases to exist and the velocity with which it travels are some of the extractable temporal features.

## 4 Experimentation

For validating the technique proposed, two videos of sea are used. The first is a video of a turbulent sea shot from near the seaside town of Cannon Beach in the USA, acquired from [11]. The second is a video of a calm sea captured from Marine Street Beach, California, obtained from the source [12].

### 4.1 Music Composed from Turbulent Sea

The turbulent sea video is broken into frames and carefully scrutinizing them reveals that the waves are quite ferocious and that tracking each and every wave is impossible. Thus, collecting their temporal features would prove to be a futile task but they possess significant spatial information. Wave-demarcated frames are obtained from the trained LDS classifier. Each and every frame is individually analyzed, and the spatial features are collected along a particular reference line, chosen here as the vertical line joining the midpoints of the top and bottom of the frame. The first feature collected is the distance between the first two waves encountered along the reference line measured in pixels ($d$). The second feature is obtained from the thickness of the first wave encountered along the reference line ($t$), also measured in terms of pixels. The entire range of $d$ and $t$ is separated into three segments, and these two features are combined together and mapped into the seven basic musical semitones following the conditions in Fig. 5. In some cases, when the feature values of successive frames lie within the same condition zone, same semitones are generated from those consecutive frames and they are grouped together into a single instance of that particular semitone. These are played using any musical instrument. Figure 5 shows the features being extracted and encoded into semitones and the music strain composed from them.

### 4.2 Music Composed from Calm Sea

When the calm sea video is broken into frames and examined, it is seen that only one or two waves exist at the most at any time. This information signifies that not much
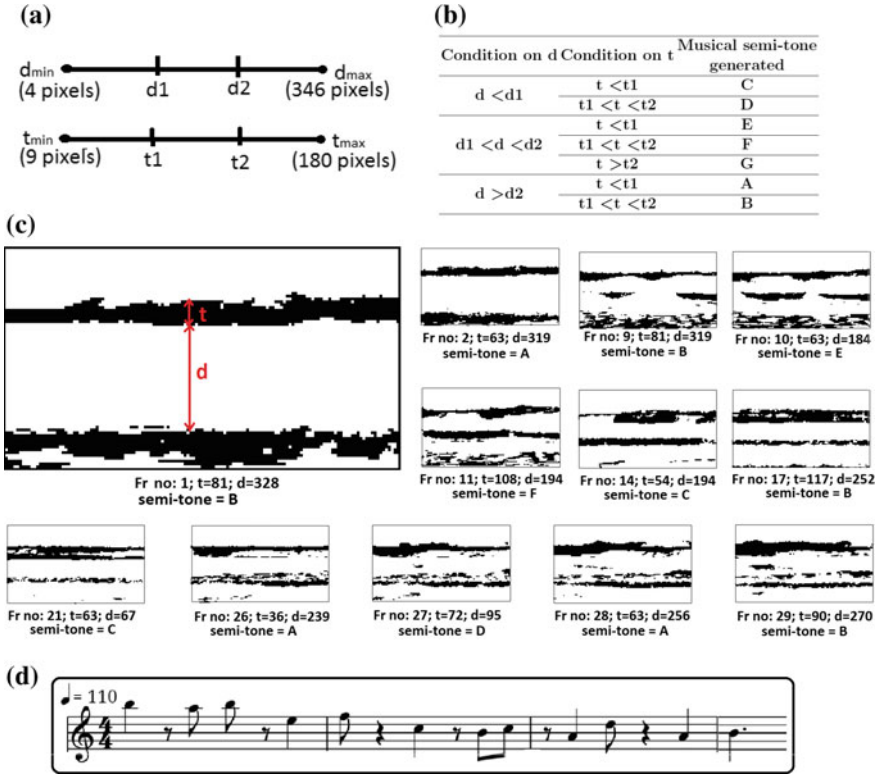
**(a)**

$d_{min}$ (4 pixels) — d1 — d2 — $d_{max}$ (346 pixels)

$t_{min}$ (9 pixels) — t1 — t2 — $t_{max}$ (180 pixels)

**(b)**

| Condition on d | Condition on t | Musical semi-tone generated |
|---|---|---|
| $d < d1$ | $t < t1$ | C |
| | $t1 < t < t2$ | D |
| $d1 < d < d2$ | $t < t1$ | E |
| | $t1 < t < t2$ | F |
| | $t > t2$ | G |
| $d > d2$ | $t < t1$ | A |
| | $t1 < t < t2$ | B |

**(c)**

Fr no: 1; t=81; d=328 semi-tone = B

Fr no: 2; t=63; d=319 semi-tone = A

Fr no: 9; t=81; d=319 semi-tone = B

Fr no: 10; t=63; d=184 semi-tone = E

Fr no: 11; t=108; d=194 semi-tone = F

Fr no: 14; t=54; d=194 semi-tone = C

Fr no: 17; t=117; d=252 semi-tone = B

Fr no: 21; t=63; d=67 semi-tone = C

Fr no: 26; t=36; d=239 semi-tone = A

Fr no: 27; t=72; d=95 semi-tone = D

Fr no: 28; t=63; d=256 semi-tone = A

Fr no: 29; t=90; d=270 semi-tone = B

**(d)**

$\flat = 110$

**Fig. 5** Turbulent sea: **a** gives the range of the features extracted and **b** gives the conditions for mapping of features extracted from wave patterns into musical semitones. **c** Features ($t$ and $d$) being extracted from frames 1 to 29 of turbulent sea and mapped into semitones. Frames that are mapped into semitones same as their successive ones are not depicted and are not used. 'Fr no' signifies frame number. **d** is the piece of music composed from the frames

spatial information could be collected, but at the same time, tracking a single wave across the frames and collecting temporal features is not so arduous a task. Wave-demarcated frames obtained as output from the LDS classifier are inspected. The first feature collected is the time of existence of each wave ($tw$). To do so, the frame number (equivalent to time) at which the existence of a wave is first detected along the mid-reference line is noted and is subtracted from the frame number at which the wave collapses (or exists no more) along the mid-reference line. The second feature collected is the distance travelled by each wave ($dw$) which is done by subtracting the starting point of the wave and the point where it collapses. The entire range of $dw$ is separated into four segments, and the range of $tw$ being less is segmented into two segments, and the two features are jointly mapped into the seven basic musical semitones (making sure consecutive like-valued semitones are written only once)
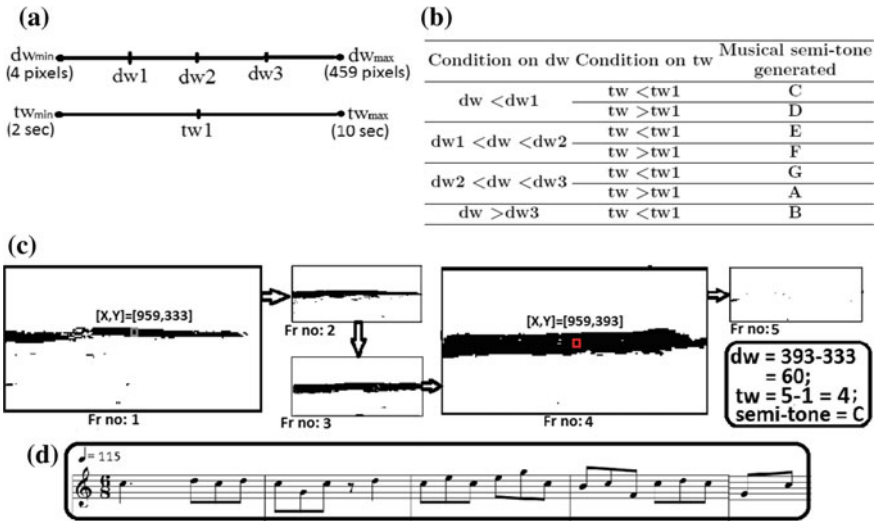
**(a)**

dwmin (4 pixels) • — dw1 — dw2 — dw3 — • dwmax (459 pixels)

twmin (2 sec) • — tw1 — • twmax (10 sec)

**(b)**

| Condition on dw | Condition on tw | Musical semi-tone generated |
|---|---|---|
| dw < dw1 | tw < tw1 | C |
| | tw > tw1 | D |
| dw1 < dw < dw2 | tw < tw1 | E |
| | tw > tw1 | F |
| dw2 < dw < dw3 | tw < tw1 | G |
| | tw > tw1 | A |
| dw > dw3 | tw < tw1 | B |

**(c)**

[X,Y]=[959,333]  Fr no: 2  Fr no: 1  Fr no: 3  [X,Y]=[959,393]  Fr no: 4  Fr no:5

dw = 393-333 = 60; tw = 5-1 = 4; semi-tone = C

**(d)** ♩= 115

**Fig. 6** Calm sea: **a** gives the range of the features extracted and **b** gives the conditions for mapping of features into musical semitones. **c** Features ($tw$ and $dw$) of the first wave being extracted from frames 1 to 5 and mapped into semitone, and 'Fr no' signifies frame number. **d** is a sample piece of music composed

as in Fig. 6. The procedure adopted to extract the features and encode them into semitones along with a sample strain of music composed is depicted in Fig. 6.

## 5 Results and Conclusion

The LDS classifier employed to demarcate waves is found to give a percentage of success of 100% in the case of training, for both the turbulent and calm sea datasets. In the case of testing, in both cases, 25 randomly chosen frames are manually compared with the wave-demarcated output frames obtained from the trained LDS and are found to be appropriately classified, with every wave clearly highlighted. These high percentages of success along with the fact that the LDS classifier requires the storage of only four coefficients (by using linear phase property while designing the LDS with seven coefficients) and has no need for dimensionality reduction techniques stand testimony to the fact that the LDS is highly storage and computation efficient without compromising the performance.

In order to evaluate the work done, three evaluation experiments are done by enlisting the aid of 50 people by means of Google form [13]. In the first two tests, they are asked to identify the machine-composed music piece from an already existing human-composed music clipped to the same duration, all of them being played by the same musician, using the same electronic keyboard so as to create a platform for

fair evaluation. The fact that 58 and 60% chose wrongly in the two tests indicates that the machine compositions are as musical as human ones and that they are unable to distinguish between them. In the third experiment, random numbers are encoded into musical semitones following the same procedure as the one in this proposal and played in the same instrument, by the same person. In order to analyze the significance of this proposal's contribution to the field of music composition, the subjects are asked to compare this audio with a machine-generated piece and testify to how vastly both vary and that random numbers are incapable of generating music. Their rating of 91.6% reveals that the project is worthy of contributing much to the music composition domain.

# References

1. Muoz E, Cadenas JM, Ong YS, Acampora G (2016) Memetic music composition. IEEE Trans Evol Comput 20:2202–2213
2. Santos A, Arcay B, Dorado J, Romero J, Rodriguez J (2000) Evolutionary computation systems for musical composition. Proc Acoust Music: Theory Appl 1:97–102
3. Engels S, Chan F, Tong T (2015) Automatic real-time music generation for games. In: Proceedings of the eleventh AAAI conference on artificial intelligence and interactive digital entertainment
4. Roig C, Tardn Lorenzo J, Barbancho I, Barbancho AM (2014) Automatic melody composition based on a probabilistic model of music style and harmonic rules. Knowl-Based Syst 71:419–434
5. Merwe A, Der V, Schulze W (2011) Music generation with markov models. IEEE Multimedia 18:78–85
6. Conklin D (2003) Music generation from statistical models. In: Proceedings of the symposium on artificial intelligence and creativity in the arts and sciences, pp 30–35
7. Miranda E, Biles J (2007) Evolutionary computer music. Springer, Berlin
8. Pearce M, Wiggins G (2001) Towards a framework for the evaluation of machine compositions. In: Proceedings of the AISB01 symposium on AI and creativity in arts and science, pp 22–32
9. Florintina C, Gopi ES (2017) Constructing a linear discrete system in kernel space as a supervised classifier. IEEE WiSPNET 2017 conference (accepted)
10. Gopi ES, Palanisamy P (2014) Maximizing Gaussianity using kurtosis measurement in the kernel space for kernel linear discriminant analysis. Neurocomputing 144:329–337
11. Breaking waves—1 hour of beautiful Pacific Ocean waves in HD. Retrieved from https://www.youtube.com/watch?v=e_2t83JsbGI. Accessed on 01 May 2017
12. Relaxing 3 hour video of California Ocean waves. Retrieved from https://www.youtube.com/watch?v=zmPzbZVUp3g&t=2s. Accessed on 01 May 2017
13. Evaluation of music composed through computational intelligence inspired by sea waves. Retrieved from https://docs.google.com/forms/d/e/1FAIpQLSd7LOai_W6mSZcDN82rGSBIT8zXodtuACaVIaB2tOBlOQCs8A/viewform?c=0&w=1