# MAP SEEKING CIRCUITS: A NOVEL METHOD OF DETECTING AUDITORY EVENTS USING ITERATIVE TEMPLATE MAPPING

*B. Jerry G. Gregoire and Robert C. Maher*

Department of Electrical and Computer Engineering, Montana State University, Bozeman MT 59717
jgregoire@ece.montana.edu , rob.maher@montana.edu

## ABSTRACT

This paper reports on a new algorithm to detect the presence of a known acoustic signal in an unknown source. The algorithm, Map Seeking Circuits, has been successfully used in the visual domain. The algorithm seeks to find an appropriate transform that will match a stored template to an unknown signal. The algorithm uses superposition to significantly reduce the computational complexity of searching for a given feature in a signal. This results in a linear computational increase rather than an exponential increase as the complexity of the signal increases. The algorithm was tested with a corpus of six instruments. Results varied from 66% for the piano to 94% for the horn.

***Index Terms*****—** map seeking circuits, acoustic, detection, template matching

## 1. INTRODUCTION

The human auditory system is quite adept at recognizing sounds. It appears that the auditory system uses both source separation and recognition to accomplish this task. Acoustic source separation often falls in the category of *Computational Auditory Scene Analysis* (CASA). Although progress has been made in CASA research, nothing has come close to the capability of the human auditory system.

In the past several years, research has focused on sound identification or classification. Most of the techniques proposed rely on traditional pattern classification techniques which require a clean signal with little noise.

This work proposes using a new algorithm with the goal of detecting acoustic events in a noisy background. In this context noise is defined as any additive signal in addition to the sought after target. This paper reports on work showing that the algorithm is capable of identification of a clean target. Subsequent work will test the algorithm with noise added to the input signal.

Classification and detection techniques fall into two major categories: feature based classifiers and template matching. Feature based classifiers extract a feature vector from a signal and typically uses a clustering algorithm such as k-means to discriminate between groups for classification or between individuals for detection. Template methods use a known representation of a target signal and attempt to match it to a pattern.

A common problem for template matching techniques is the computational complexity that increases exponentially with the dimensionality of the data set. To overcome this, transforms are often used to produce invariance along one or more dimensions [1].

Arathorn proposed a novel template matching technique, *Map Seeking Circuits* (MSCs)*,* to overcome the combinatorial explosion of template matching without the need to define invariant transforms [2]. A MSC seeks to find an appropriate set of transforms that map a stored template to an unknown signal. The algorithm uses superposition with an iterative matching process to converge on the best set of transforms that map a template to a target in an input signal.

A MSC is comprised of one or more layers and a set of templates. Each layer represents a dimension and an associated transform such as translation, scale or rotation. The algorithm performs a set of transforms at each layer and sums the result. The result is then sent to the following layer where the process is repeated for another dimension. The algorithm depends on *The Ordering Principle of Superposition* [3]. The principle states that if matches are computed between a pattern, *A*, and a superposition of a set of patterns, the match will be greatest for the pattern within the superposition that is most like *A*. The use of superposition reduces the computational complexity from exponential growth to linear growth, thus making the problem tractable.

MSCs employ a nonlinear competition function to cull out the poorer transforms. The process is iterative and continues until it convergences to a solution The ordering principle of superposition ensures that the MSC finds the best transform set that maps the template to the target within the input signal.

Work has been submitted showing that the MSC algorithm will converge to either a set of unique transforms, one for each layer, or a null condition which indicates that a

1-4244-0535-1/06/$20.00/©2006 IEEE

mapping of the template to the test signal is not possible with the given set of transforms [4].

In this paper we expand on previous work that uses the MSC concept for acoustic signals, *Acoustic Map Seeking Circuits* (AMSCs). Previously, we demonstrated a single layer AMSC using the amplitude of an instrument's spectrum to identify an input signal [5]. The template and signal were limited to the sustained portion of the signal. In this paper we demonstrate a three layer AMSC that uses amplitude, time, and frequency transforms for each layer. We also allow the signal to have an attack and decay portion. Research has shown that the attack portion of an acoustic signal is important for its recognition by humans [6, 7, 8] and is valuable for identification of instruments by automatic means [9, 10, 11].

The AMSC uses a gammatone filter bank to create a time versus frequency representation of both the input signal and the template or *gammagram*. The gammagram is a biologically inspired frequency versus time representation of an acoustical signal. Since the bandwidth of a gammatone filter bank increase with the filter's center frequency, its use also leads to a more compact representation along the frequency dimension.

An instrument's template is created by combining several gammagrams of adjacent semitones produced by the instrument. This results in a characteristic surface that represents the instrument's resonances and temporal evolution in the time-frequency plane. The AMSC then uses simple shifts along the time, amplitude and frequency axes to align the template with the test signal's gammagram. The algorithm requires at least one transform along each dimension to have a minimum match value and a transform that consistently produces a better match throughout the iterative process. If either of these conditions fails the algorithm produces a null condition. A null condition indicates that the AMSC failed to find a possible mapping between the target and the template.

## 2. MAP SEEKING CIRCUITS

The MSC concept is shown in Figure 1. The MSC has two paths, a forward path and a backward path. The forward path transforms the input along predefined dimensions. If a set of the transforms match the input to one of several stored templates, the most promising transforms are enhanced on the backward path by virtue of a competition function. After several iterations, the MSC has one of two possible outcomes: the best template is mapped via the chosen transforms to an object in the input, or no transform set is found. In the latter case this is the null condition; none of the templates were found in the mixture.

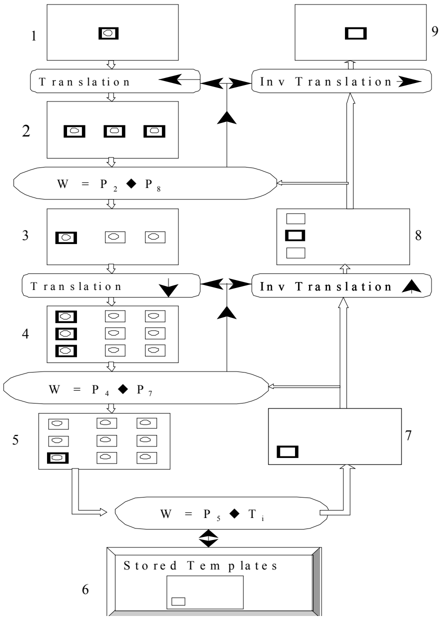

**Figure 1: Map seeking circuit topology**

Figure 1 illustrates a MSC with two transform sets and one template. The left hand side of the diagram is the forward path. Here the input is transformed first horizontally, shown in panel two, and then vertically as shown in panel 4. In this example, a horizontal shift to the left and a vertical shift down match the transformed square to the template which is also a square.

On the backward path, the template is compared to panel 4 and is then inverse transformed where it is then compared to panel 3. One final inverse transform places the template in the same position of the square in the original input, but without the circle.

As the iterations progress, the poorer transforms are attenuated with respect to the better ones. Eventually, the algorithm forces all transforms to zero except for the best transform along each dimension. At this point the algorithm has successfully mapped the template to the mixture. If the algorithm can not find a mapping between the target and the template, then all the transforms are driven to zero resulting in a null result.

The MSC algorithm will always converge to either a set of unique transforms or a null condition where all the transforms in each layer are driven to zero. In the case that a unique set of transforms are found, the similarity of the input and the inverse transformed template must exceed a threshold for a positive mapping to result. The threshold is set by comparing the similarities of several positive inputs to negative inputs during a training session. Ideally, there will be a large discrepancy between the two. If there is overlap, then amount of false rejection errors can be traded for false acceptance errors depending on which is considered more valuable for a given application.

## 3. METHOD

### 3.1. Data preparation

The input and template AMSC objects are three dimensional representations of the sound along the time, amplitude and frequency axes. To convert a sound into an AMSC object, the sound was first band limited to 8 kHz and then spectrally decomposed using a gammatone filter bank. The magnitude of each filter bank output, or bin, was then found by calculating the absolute value of its analytic signal. To reduce the size of the data set, the outputs were then downsampled by 10. Finally, the data was converted to a three dimensional representation that is equivalent to a waterfall plot. The total time of each sample was 62.5 ms.

### 3.2. Templates and targets

Both the template and target are prepared as described in Section 3.1. The template differs from a target in that it is a composite of 12 semitones spanning an octave. It is also complied with a set of two or three instances of the instrument, depending on what was available.

Two methods were used to create the templates. The first, *Aver*, averaged the instrument's partials along the amplitude axis. Thus, it is effectively is a surface described in the three dimensional space.

The second method added the partial tracks in three dimensions to create an aggregate representation of the template, *Aggr*. Whereas the first method resulted in a single track in the amplitude and frequency plane, the second method allowed each template instrument to have its own track. This created a surface as the first method did but with a finite volume in the amplitude dimension.

The target was assumed to be a single note within the octave used to create the template. As such is has a discrete set of contours that describe the amplitude of each partial as a function of time.

### 3.3. Transforms

The transforms used in this paper are simple shifts along each of the axes: Time, Amplitude, and Frequency. The spectrum of an instrument is largely determined by its resonance characteristics. The goal therefore, is to match the contours of the target's partials to the surface of the template.

The time transform varied +/- 23.4 ms in 1.6 ms increments. The amplitude transform range was chosen to ensure that it fully bracketed the average RMS power calculated over the duration of the signal, typically -10 dB to 30 dB. The resolution of the amplitude transform was 1dB. The frequency transform had a more restricted range of +/- 1 bin. Another set of tests were ran with no shift allowed in the frequency dimension.

### 3.4. Data

The templates were derived from the RWC instrument database [12]. The RWC database has three instances for most instruments played at a variety of dynamic levels and styles. The sounds used in this paper were limited to a common dynamic level, forte, and were non vibrato. The semitones of the fourth octave, C4 to B4, comprised the template.

The target samples were obtained from the musical instrument recordings compiled by the University of Iowa Experimental Music Studios [13]. They were limited to the same dynamic level and playing style as the templates. As with the template the targets were taken from the fourth octave.

Six instruments were chosen to encompass the string, brass, and woodwind families. They were the alto sax, flute, French horn, oboe, piano, and violin.

### 3.5. Experiments

Four variations of the experiment were performed as shown in Table 1. Each variation consisted of a matrix using one of the six templates and the six target instruments as test input. Additionally, each instrument provided five notes, across the fourth octave.

| Experiment Variation | +/- 1 Bin | No Shift |
|---|---|---|
| Aggregate | Aggr +/-1 | Aggr 0 |
| Average | Aver +/-1 | Aver 0 |

**Table 1:** Experimental variations.

Six tests were run using the templates of the six instruments. Each test compared six recorded notes of each instrument to the template in a given test. The musical pitches used were C4, D4, E4, F4, A4 and B4. The flute was slightly different because one of the samples used for the template did not contain the entire fourth octave. The notes used for the flute targets were C4, D4, E4, F4 and G4. The error rate was calculated for each combination of the experiments listed in Table 1 and the templates. The experiment that provided the best error rate for each of the six templates is listed in the results section.

The error rate was calculated using

$$P_T = (P_P E_P + P_N E_N)/N,$$

where $P_P$ and $P_N$ are the probabilities of a positive and negative target respectively. $E_P$ is the number of false rejections of a positive target and $E_N$ is the number false acceptances of a negative target. N is the total number of notes or targets in the corpus.

A positive target is a sample taken from the same type of instrument that was used to create the template. Conversely, a negative target is one that is from a different instrument than the template instrument.

Each test had 30 targets of which five were positive and 25 were negative. Thus,

$$P_P = 5/30,\ P_N = 25/30 \text{ and } N = 30.$$

## 4. RESULTS

The discrimination threshold mentioned in Section 2 was a hard decision boundary. That is, if a positive target resulted in a value less than the threshold it was rejected. Conversely, if it was above the threshold, it was accepted and visa versa for a negative target. No attempt was made to judge the distance between the final reconstructed similarity value and the threshold.

Tables two and three tabulate the total error as a function of the number of positive targets rejected. They also list the experiment variation that resulted in the minimum error.

| Template | Negative Acceptances | Total Error $P_E$ (%) | Best Variation |
|---|---|---|---|
| AS | 11 | 31 | Aggr 0 |
| FL | 6 | 17 | Aggr +/-1 |
| HN | 2 | 6 | Aver +/-1 |
| OB | 10 | 28 | Aver 0 |
| PN | 13 | 36 | Aggr +/-1 |
| VN | 6 | 17 | Aver 0 |

**Table 2:** AMSC incorrect acceptances with threshold set for zero positive rejections.

| Template | Negative Acceptances | Total Error $P_E$ (%) | Best Variation |
|---|---|---|---|
| AS | 10 | 28 | Aggr 0 |
| FL | 5 | 14 | Aggr +/-1 |
| HN | 2 | 6 | Aver +/-1 |
| OB | 4 | 12 | Aver 0 |
| PN | 13 | 37 | Aggr +/-1 |
| VN | 5 | 14 | Aver 0 |

**Table 3:** AMSC incorrect acceptances with the threshold set for one positive rejection.

Table 4 is a confusion matrix of the false acceptance of a negative target when the threshold is set for zero false rejections of the positive targets. Each row is a different template, and the columns represent the targets. The totals in the rightmost column are the incorrect acceptances of a target by a given template, while the totals in the bottom row are the incorrect rejections of each target.

|  | AS | FL | HN | OB | PN | VN | TOTAL |
|---|---|---|---|---|---|---|---|
| AS | 0 | 5 | 2 | 1 | 2 | 1 | 11 |
| FL | 0 | 0 | 2 | 0 | 3 | 1 | 6 |
| HN | 0 | 1 | 0 | 0 | 1 | 0 | 2 |
| OB | 1 | 2 | 1 | 0 | 2 | 4 | 10 |
| PN | 4 | 3 | 4 | 0 | 0 | 2 | 13 |
| VN | 2 | 3 | 0 | 1 | 0 | 0 | 6 |
| TOTAL | 7 | 14 | 9 | 2 | 8 | 8 | 48 |

**Table 4:** Errors with the threshold set for zero positive rejections.

## 5. DISCUSSION

The instruments tested fell into three categories. In the top category is the horn with an error of just 6%. The flute and the violin fall in the middle with an error of 17%. The AMSC performed worst with the oboe, alto sax and piano. This was somewhat surprising. We expected the representation to adequately capture the formants produced resonant structure of the wind instruments. The violin performing better than the oboe or alto saxophone was counter intuitive. It is possible that the sample window used, 62.5 ms, was simply too short to capture the spectral characteristics of the woodwinds which is predominate in the sustained portion of the sound. The impetus for using a short sample was primarily to focus on the attack potion of the signal which is assumed to be more invariant that the sustained or decay portions.

While the flute template was best in correctly tagging the flute targets, the other templates falsely accepted a flute target more than any other instrument. This is seen in the total incorrect rejections in the bottom row of Table 4. This may be due to the relatively pure sinusoidal characteristics of the flute partials.

The oboe was just the opposite. The oboe template incorrectly tagged another instrument as an oboe 10 times, yet the other templates rarely accepted the oboe target.

Overall, the error rates shown in Tables 2 and 3 compare favorably to previous work. Fujinaga using a variety of weighted moments obtained error rates of 8% for the horn, 30% for the flute and more than 60% error for the oboe and violin [14].

Brown used cepstral coefficients and was able to discriminate between an oboe and saxophone with an error of 10% and 4% respectively [15]. Although the overall performance of the oboe and saxophone had about 30% error, when they are compared against each other, the AMSC performed quite well. It incurred only one error when comparing the horn to the oboe template and produced no error when the horn template was presented with an oboe.

It was also quite surprising that there was no clear winner between the four variations generated between the two different template representations and the frequency transform. The frequency transform was included because some samples such as the oboe were found in preliminary work to have the partials shifted by one gammatone bin. It was thought that allowing a minor variation in frequency would help the performance of the AMSC when presented with an oboe target. In fact just the opposite was true. The oboe template performed best when there was no transform in the frequency.

It is not clear why none of the variations proved better overall. It is possible that the characteristics of some instruments tend to favor one variation over another. In this case the better performing variation would be included in a portfolio that would be loaded whenever that particular instrument or class of sounds was sought to be detected.

## 6. FUTURE WORK

The eventual goal of this effort is to develop a robust technique to detect a sound in the midst of noise. Identification of a pure sound as demonstrated in this paper is the first step. Ongoing work includes testing the current implementation with varying amounts and types of noise.

The representation can be further refined to include additional characteristics that are valuable in sound discrimination. Finally, a more compact representation of the data will lead to a faster and more versatile algorithm.

## 7. REFERENCES

[1] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*: Acedemic Press, (Elsevier Science), 1999.

[2] Arathorn, *Map Seeking Circuits: A Computational Mechanism for Biological and Machine Vision*. Stanford: Stanford Press, 2002.

[3] Arathorn, "Recognition Under Transformation Using Superposition Ordering Property," *IEE Electronics Letters*, vol. 37, pp. 164-166, 2001.

[4] T. Gedeon and D. Arathorn, "Convergence of Map Seeking Circuits," *Journal of Mathematical Imaging and Vision*, 2006. (submitted)

[5] Gregoire, B.J. and Maher, R.C., “Harmonic Envelope Detection and Amplitude Estimation Using Map Seeking Circuits,” Proc. IEEE International Conference on Electro Information Technology (EIT2005), Lincoln, NE, May, 2005.

[6] K. W. Berger, "Some Factors in the Recognition of Timbre," *Acoustical Society of America*, vol. 36, pp. 1888 - 1891, 1964.

[7] E. L. Saldanha and J. F. Corso, "Timbre Cues and the Identification of Musical Instruments," *Acoustical Society of America*, vol. 36, pp. 2021 - 2026, 1964.

[8] M. Clark, "Perturbations of Synthetic Orchestral Wind-Instrument Tones," *Acoustical Society of America*, vol. 41, pp. 277-285, 1967.

[9] I. Kaminskyj and Voumard, "Enhanced Automatic Source Identification of Monophonic Musical Instrument Sounds," presented at Intelligent Information Systems, Adelaide, Australia, 1996.

[10] K. Eronen, "Musical Instrument Recognition Using Cepstral Coefficients and Temporal Features," presented at IEEE ICASSP, Istanbul, Turkey, 2000.

[11] I. Fujinaga and K. MacMillan, "Realtime Recognition of Orchstral Instruments," presented at ICMC, 2000.

[12] M. Goto, "Development of the RWC Music Database," presented at ICA, 2004.

[13] Iowa University, "Electronic Music Studios." http://theremin.music.uiowa.edu/MIS.html

[14] I. Fujinaga, "Machine Recognition of Timbre Using Steady-state Tone of Acoustic Musical Instruments," presented at Intl Computer Music Conference, 1998.

[15] J. C. Brown, "Computer identification of musical instruments using pattern recognition with cepstral coefficients as features," *Journal of Acoustic Society of America*, vol. 105, pp. 1933-1941, 1999.