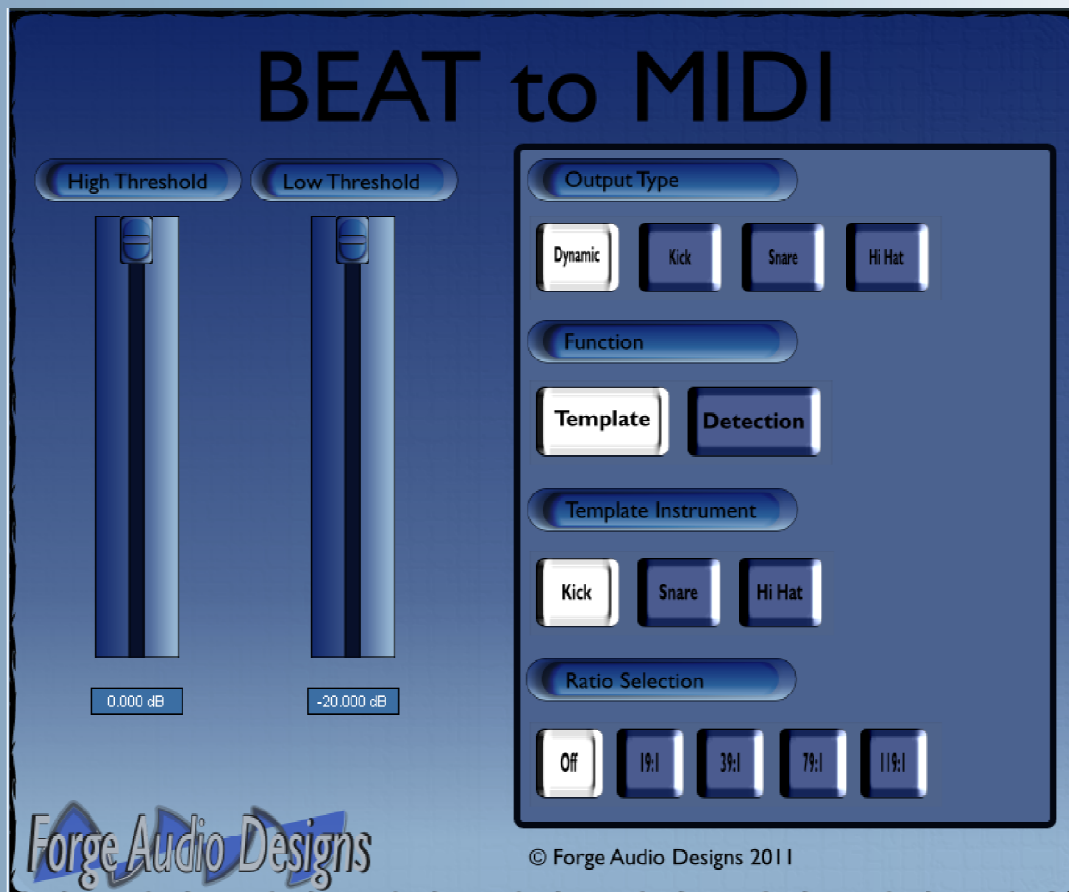


Forge Audio Designs BEAT to MIDI V1.0.0 Quick Start Guide



Forge Audio Designs

BEAT to MIDI V1.0.0

Quick Start Guide

Welcome to the BEAT to MIDI Quick start guide, here you will find some basic tips on installation and operation.

Introduction

BEAT to MIDI has been developed as a low cost VST BEAT detection solution. It offers a high resolution detection algorithm that offers BEAT detection based upon defined parameters.

Detections are output as MIDI notes based upon the General MIDI drum map as Kick, Snare or Hi Hat. In addition to this, detections can be made dynamically via the pattern matching template system. You may create a template based on an instrument and BEAT to MIDI will dynamically detect this type of instrument.

BEAT to MIDI V1.0.0 is the first release and will be continually updated in the future; as such some features will be updated to offer increased performance and efficiency.

Requirements

- Windows 7
- VST 2.4 compliant host (32-bit only)
- 1024 x 768 display
- 1 GB Ram
- 1 GHz CPU (Average 5% ASIO load on Dual Core 2.4 GHz system)

Hosts

Please note that BEAT to MIDI has only been tested in the following VST hosts:

- Cubase 5 (32-Bit)
- Cubase 6 (32-Bit)

Although compatibility between VST hosts should be applicable it is not guaranteed.

Installation

BEAT to MIDI is currently supplied as a .dll file and as such must be installed manually rather than via an installation wizard.

To install BEAT to MIDI please navigate to your chosen host DAWs VST plug-in folder. For example, when using Steinberg Cubase the VST plug-in folder can be found at the location:

Program Files / Steinberg / Cubase x.x / VST Plugins

Paste or move the BEAT to MIDI .dll to this location and ensure that your host re-scans its VST folder on initialization.

Please note that currently, BEAT to MIDI is Windows PC only compatible.

The Interface

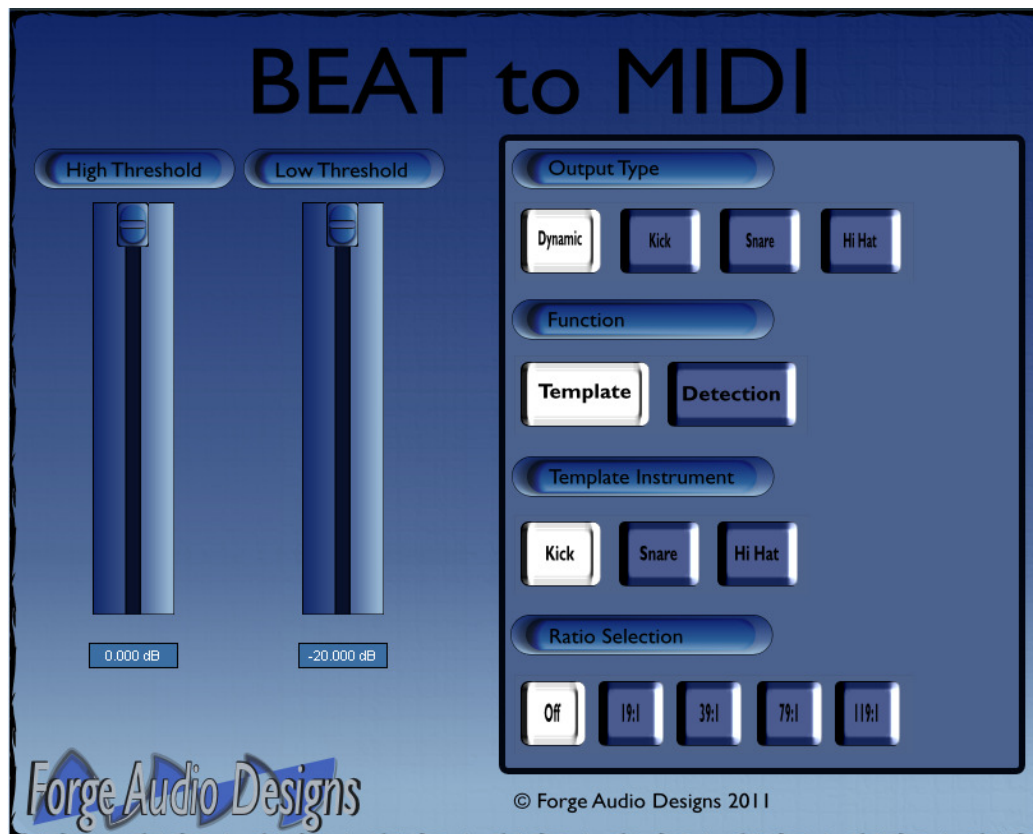


Fig 1. The BEAT to MIDI Interface

BEAT to MIDI consists of 6 controls for detection and dynamic pattern matching detection.

High Threshold: This value is used to set the amplitude level that incoming audio must exceed for it to be declared as a detected transient or hit point. This allows some transients to be excluded i.e. Snare bleed through on kick drum microphone. The high threshold range is between 0 and 1.

Low Threshold: The low threshold value is used to 'reset' the trigger. The incoming audio must fall below this value before a new transient can be detected. Its range is from 0 to 0.1.

Output Type: This control sets the MIDI note that detected transients will be output to (Based upon the MMA GM MIDI drum map):

- Dynamic – Dynamic detection via pattern matching
- Kick – Kick Drum (36)
- Snare – Snare Drum (38)
- Hi Hat – Hi Hat (42)

Function: The function control allows you to switch between the 'detection' and 'template creation' method:

- Template creation
- Detection

Template Instrument: When set to template creation the user can create templates for instrument matching. This control allows the user to define what instrument the template will be linked to:

- Kick Drum
- Snare Drum
- Hi hat

Ratio Selection: Research has shown that there are certain ratio's between the high and low thresholds that offer the best detection (highest number of correct detections, least number of errors). These ratio values are those that proved to be the most successful, they can be used as a permanent setting or to offer guidance. Please note that when using a ratio setting, control to the low threshold is removed, switch the selection to 'Off' to regain independent control of the low threshold. The switch offers 5 differing states:

- Off – High and Low threshold faders can be controlled independently by the user
- 19:1 – The user may only control the high threshold, the value of the low threshold fader is scaled to a value of 19:1
- 39:1 – The user may only control the high threshold, the value of the low threshold fader is scaled to a value of 39:1
- 79:1 – The user may only control the high threshold, the value of the low threshold fader is scaled to a value of 79:1
- 119:1 – The user may only control the high threshold, the value of the low threshold fader is scaled to a value of 119:1

The Pattern Matching System

Although BEAT to MIDI can be used to detect transients and simply output the MIDI to single notes it is sometimes useful to have a dynamic detection system; so that when analysing a drum loop that contains more than one instrument, these differing detection may be output to the appropriate MIDI note.

The best way to implement this is to create templates for each instrument. When a template is created, BEAT to MIDI stores the template for that instrument. When the incoming audio matches that template, the MIDI note that is output is mapped accordingly.

To use the pattern matching system ensure function is set to 'Template creation' and then use the 'Template Instrument' function to define what drum the template will be associated with (*see 'the interface'*).

Once you have this set appropriately, simply play the audio for that single drum sound and BEAT to MIDI will automatically create a template for it. Please note, it is advised that you set your required threshold settings appropriately for your project before creating the template as changes in the thresholds after the template is created can lead to false detections.

Implementation and Use

Please note that this guide uses Steinberg Cubase 5 for all examples.

Once installed and your host has been started you will need some test audio data. BEAT to MIDI is designed for use with audio drum loops so it would be an idea to load such a loop into your DAW from a library or alternately, generate your own loop via a Drum VST and re-import it to your project as an audio file.

Once you have this file, you may place BEAT to MIDI as an Insert VST on the audio track.



Fig 2. Steinberg Cubase 6. Using BEAT to MIDI on an audio track

You will now see that BEAT to MIDI appears as a VST in your project. The next stage is to create a MIDI track to receive MIDI output data from BEAT to MIDI. To do this, create a new MIDI track and choose the 'output' of BEAT to MIDI as the 'input' for the new MIDI track. Ensure that the MIDI channel is set to 1.

N.b. This procedure may vary dependant on host, please consult your manufactures instruction guide for more information



Fig 3. Steinberg Cubase 6. MIDI track

Now you are ready to capture some MIDI data. Set your MIDI track to record and experiment to find the best settings for detection (ensure that function is set to detection), this may take some trial and error. To aid in the detection setup process you may use the ratio settings, these offer different ratio strengths between the high and low thresholds. These ratios offer the best results for detections but are only suggested values. Sometimes they may work well for your project; at other times they should be used as a starting point.

Below you can see that we have now successfully captured the transient data, in this instance, all the notes have been output to a single MIDI note. However, if using the dynamic detection method then notes may be output on different keys.

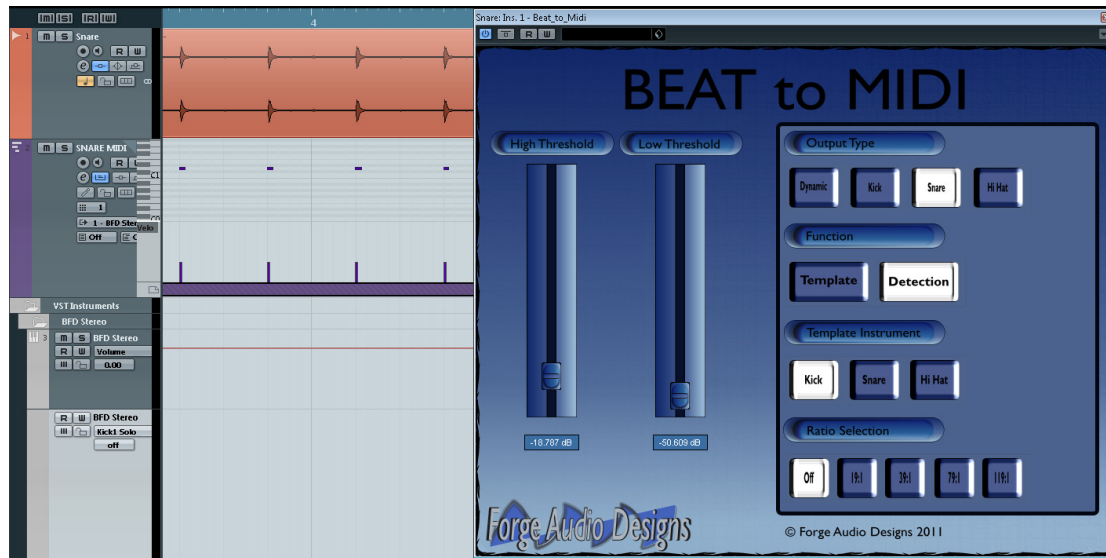


Fig 4. Steinberg Cubase 6. Completed detection

Hints and Tips

1. When creating a template for pattern matching you should ensure you only play the audio for the drum you wish to create a template for. If you wish to capture another drums template then ensure you move the slider to a different drum on 'Template Inst' otherwise settings will be over-written.
2. When using the pattern matching (dynamic detection) method it is suggested that you find the appropriate threshold values for detections before creating the template. As altering the threshold values after creating the template may lead to the pattern matching system not functioning correctly.
3. Always ensure your DAW playback begins a bar before the transient(s) you wish to detect or create templates for.
4. It can sometimes take some fine balancing to get the best detection. You may find that in some cases you will get MIDI 'trains' of multiple detections. If this is the case, try lowering the 'Low Threshold' first. Sometimes some balancing is needed between the high and low threshold values. Use the Ratio system to define a good starting point, using harsher ratio settings if necessary.

Thank you!

We hope you enjoy BEAT to MIDI!

About

BEAT to MIDI is a product of Forge Audio Designs



© Forge Audio Designs 2011

www.forgeaudiodesigns.com

dan@forgeaudiodesigns.com

Original C++ VST SDK by Steinberg

VST is a trademark of Steinberg Media Technologies GmbH

- VST PlugIn Technology by Steinberg,
- VST PlugIn Interface Technology by Steinberg Media Technologies GmbH

