“Simple” Glass Break Detector

Michael Fuhrman

10/20/20

Summary

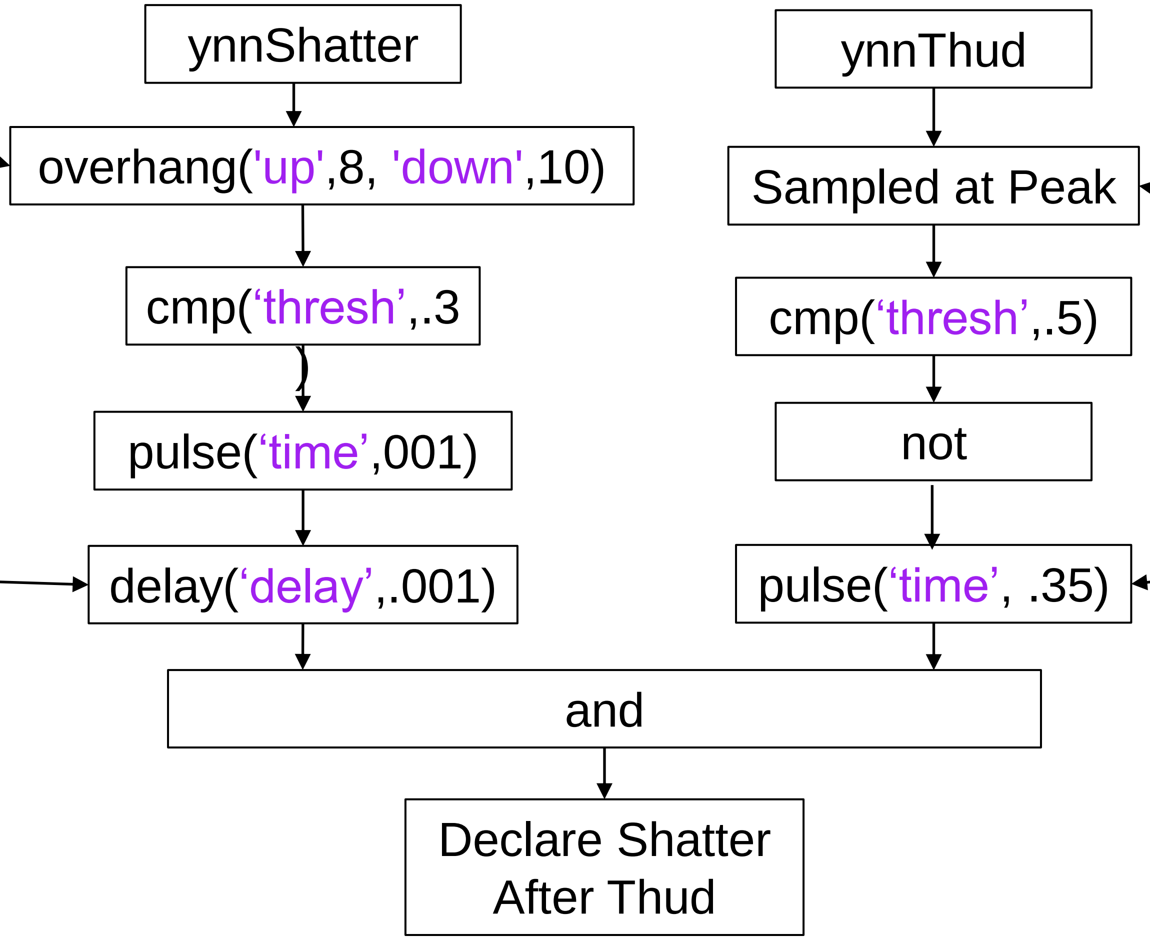
- Catching the Thud Neural Network Output at high frequency envelope peaks helps to suppress false positives while risking more false negatives.
- Suppressing Thuds after Shatter also brings improvement.
- Need to review why the simple (Shatter After Thud) detector is missing detections, but why suppressing Thuds after Shatter brings improvement.

ASPINITY

“Simple” Glass Break Chain

Shatter should have some extent in time

Possible small delay so shatter threshold crossing occurs after thud threshold crossing if needed



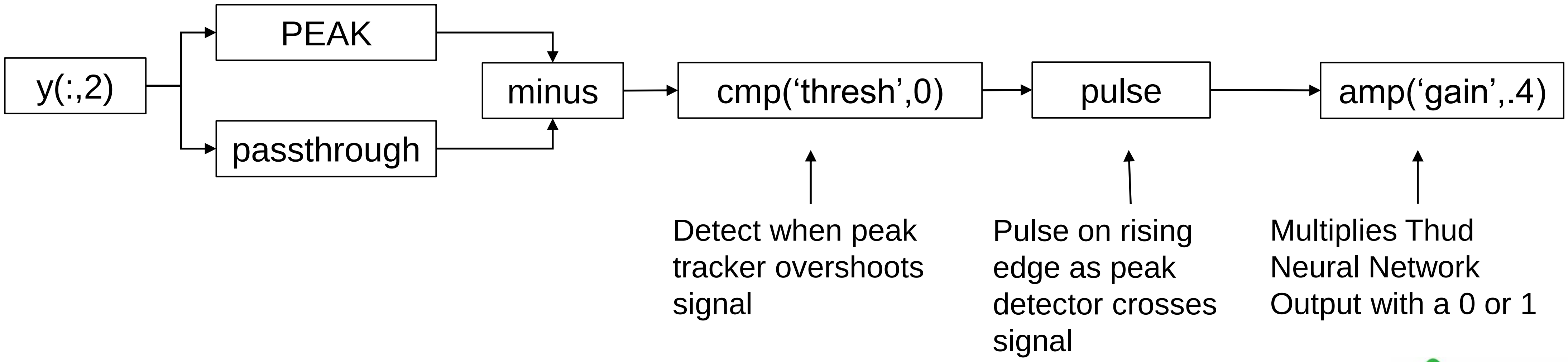
Thud Neural Network is sampled at high frequency peak (a sudden jump in signal) See next slide.

Thud umbrella extends passed onset of Shatter to ensure capture

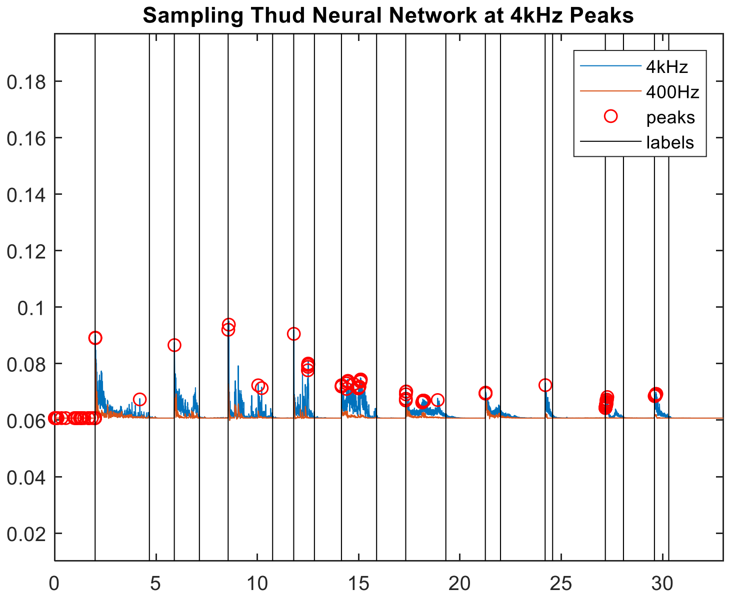
Asynchronous Peak Detector
for Sampling Neural Network Output

```
PEAK = ramp.ops.peak('atk',100, 'dec',1, 'modelVersion','PDmodelDynamic');
CMPNN = ramp.ops.cmp('thresh',.5);
AMPpt4=ramp.ideal.amp('Av',.4);
hfPeakChain = [PEAK;
       ramp.ideal.pass()] > ramp.ideal.minus() > ramp.ops.cmp('thresh',0) > …
                                   ramp.ops.pulse('time',1e-6) > AMPpt4;
peakNNChain = [hfPeakChain;
                   ramp.ideal.pass() > CMPNN] > ramp.logic.gate('type','and') > AMPpt4;
ynnThudAtPeaks = peakNNChain(t, [y(:,2), ynnThud]);
```

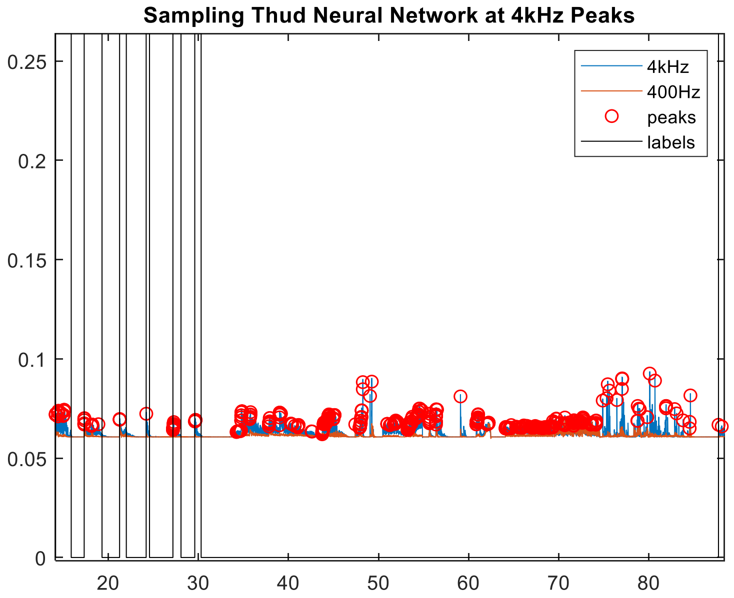
Ideally y(:,2) should not have its baseline subtracted. It should be the upper envelope of the high frequency channel.

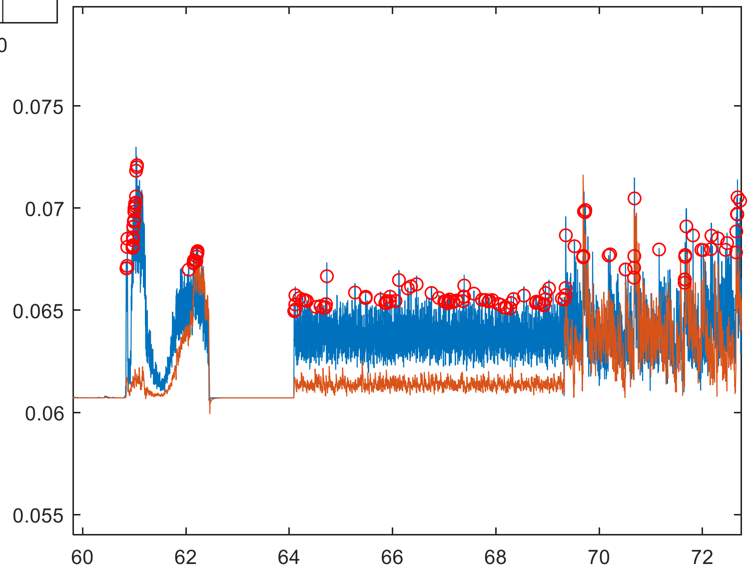


Why Focus on Signal Peaks?



4kHz peaks coincide with thuds.
400Hz peaks were sometimes missed.





But they also occur at other times.

How to get peak detector to deliver only prominent peaks?

ASPINITY

Single NN "Scratch" Glass break

The initial glass break detector collected the amplitudes and baselines of the high and low frequency envelopes along with the zero-crossing rate. A neural network trained on labeled shatter events within 200msec of onset along with background signal levels. An overhang operator

```
(Hang=ramp.ops.overhang('up',8, 'down',10))
```

applied to the neural network output filtered out events occurring over short time intervals.

TODO: How to determine overhang parameters?

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
scratch	singleNN	GB_TestClip_v1_16000	151.7189	TPR = 19 / 20	FPs = 28	83msec
scratch	singleNN	GB_TestClip_v1_16000_mixed_included	247.0156	TPR = 49 / 50	FPs = 33	68msec
scratch	singleNN	GB_TestClip_Training_v1_16000	597.461	TPR = 50 / 50	FPs = 72	121msec
scratch	singleNN	GB_TestClip_v2_16000	108.8343	TPR = 23 / 23	FPs = 16	64msec
scratch	singleNN	GB_TestClip_Short_v1_16000	26.81237	TPR = 5 / 5	FPs = 0	64msec
				146/148	149	

Comparison of Timing Constraints

Simple timing constraints, i.e., shatter after thud

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_simpleTiming	GB_TestClip_v1_16000	151.72sec	TPR = 19 / 20	FPs = 21	101msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_simpleTiming	GB_TestClip_v1_16000_mixed_included	247.02sec	TPR = 32 / 50	FPs = 34	233msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_simpleTiming	GB_TestClip_Training_v1_16000	597.46sec	TPR = 42 / 50	FPs = 49	160msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_simpleTiming	GB_TestClip_v2_16000	108.83sec	TPR = 21 / 23	FPs = 7	91.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_simpleTiming	GB_TestClip_Short_v1_16000	26.81sec	TPR = 5 / 5	FPs = 1	78.9msec
				119/148	112	

Sophisticated timing constraints, i.e., shatter after thud, suppress thud after shatter ← more TPs and FPs

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_v3Timing	GB_TestClip_v1_16000	151.72sec	TPR = 20 / 20	FPs = 19	81.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_v3Timing	GB_TestClip_v1_16000_mixed_included	247.02sec	TPR = 50 / 50	FPs = 35	58.1msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_v3Timing	GB_TestClip_Training_v1_16000	597.46sec	TPR = 48 / 50	FPs = 57	108msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_v3Timing	GB_TestClip_v2_16000	108.83sec	TPR = 23 / 23	FPs = 8	62.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_v3Timing	GB_TestClip_Short_v1_16000	26.81sec	TPR = 5 / 5	FPs = 6	77.9msec
				146/148	125	

Results: Focus on Peaks yields fewer FPs but a few misses

Simple timing constraints, i.e., shatter after thud, thud coincides with high frequency impulse

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_simpleTiming	GB_TestClip_v1_16000	151.72sec	TPR = 20 / 20	FPs = 18	100msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_simpleTiming	GB_TestClip_v1_16000_mixed_included	247.02sec	TPR = 50 / 50	FPs = 20	64.6msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_simpleTiming	GB_TestClip_Training_v1_16000	597.46sec	TPR = 47 / 50	FPs = 34	129msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_simpleTiming	GB_TestClip_v2_16000	108.83sec	TPR = 21 / 23	FPs = 4	68.9msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_simpleTiming	GB_TestClip_Short_v1_16000	26.81sec	TPR = 4 / 5	FPs = 0	76.7msec
				142/148	76	

Sophisticated timing constraints, i.e., shatter after thud, thud coincides with high frequency impulse, suppress thud after shatter

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v1_16000	151.72sec	TPR = 20 / 20	FPs = 11	81.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v1_16000_mixed_included	247.02sec	TPR = 50 / 50	FPs = 19	58.1msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_Training_v1_16000	597.46sec	TPR = 46 / 50	FPs = 33	110msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v2_16000	108.83sec	TPR = 21 / 23	FPs = 4	62.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_Short_v1_16000	26.81sec	TPR = 4 / 5	FPs = 0	76.3msec
				141/148	67	

The Wide Valid Thud Pulse AND Shatter Pulse

wideValidThudPulseChain

```
cmp('thresh',.5)          cmp('thresh',.5)
pulse('time',1e-6)        pulse('time',.2)
                          delay('delay',.05)
                          NOT
AND
NOT
pulse('thresh',.15)
```

wideValidThudPulse

A Thud which was more than 200msec after the onset of Shatter generates a 150msec wide pulse within which Shatter onset must be detected

Threshold the Thud NN
Generate a pulse

Threshold the Shatter NN
Generate a 200msec wide pulse

Delay the Shatter pulse 50msec so it doesn’t interfere with the Thud that caused it

NOT the wide pulse

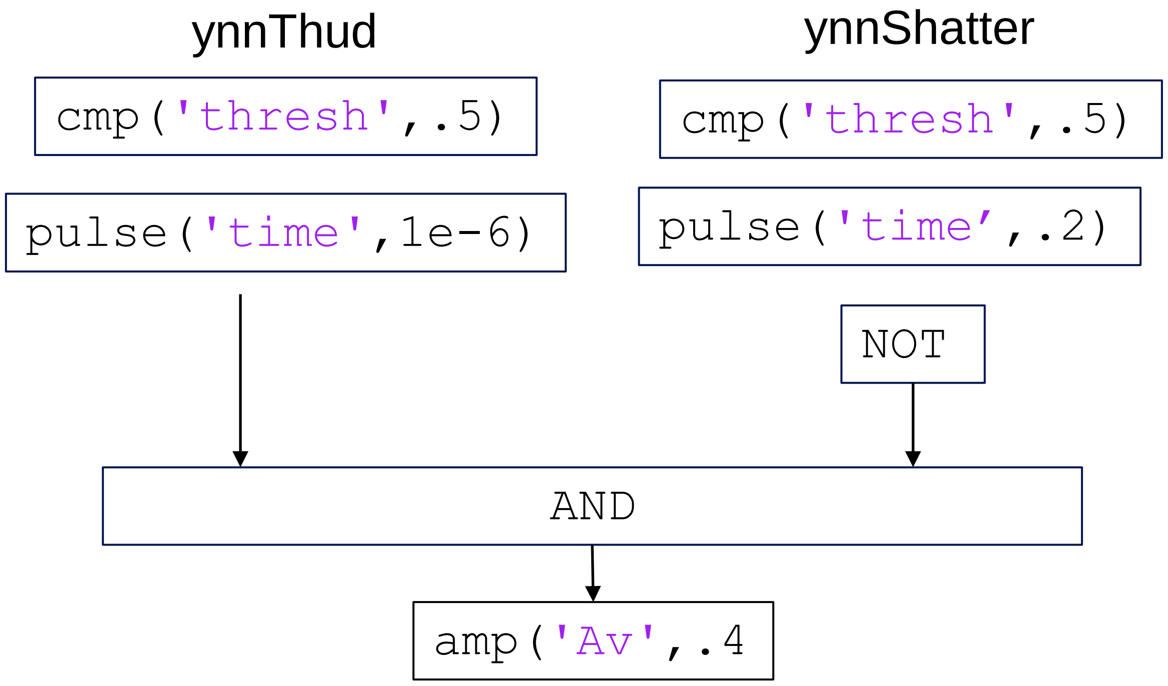
AND the result with the Thud coming after Shatter to suppress those Thuds

The remaining Thuds are valid Thuds which are widened to 150msec

ASPINITY

A Valid Thud Pulse

```
validThudPulseChain = [THUDTHRESH > PULSE0001;
         SHATTERTHRESH > ramp.ops.pulse('time',.2)> NOT] > AND > AMPpt4;
```



`validThudPulse (declareThud)`

Suppress Thuds within 200msec of the onset of Shatter

Shatter with some extent over time

```
ShatterPulseChain = SHATTERTHRESH > ramp.ops.overhang('up',1.25/UpTime,
'down',1.25/DownTime) > CMPpt5 > PULSE0001;
```

```
cmp('thresh',.5)
```

```
overhang('up',125, 'down',125)
```

```
cmp('thresh',.5)
```

```
pulse('time’,1e-6)
```

```
shatterPulse
```

The Last Step -declareShatter

AMPpt4(t,AND(t,[wideValidThudPulse shatterPulse])) to declareShatter

Comparison of Tanh and Log Results

Sophisticated timing constraints, i.e., shatter after thud, thud coincides with high frequency impulse, suppress thud after shatter

Tanh

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
glassBreakRampDevWithParameters_v3	TanhBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v1_16000	151.72sec	TPR = 16 / 20	FPs = 4	75msec
glassBreakRampDevWithParameters_v3	TanhBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v1_16000_mixed_included	247.02sec	TPR = 45 / 50	FPs = 18	65.2msec
glassBreakRampDevWithParameters_v3	TanhBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_Training_v1_16000	597.46sec	TPR = 42 / 50	FPs = 32	101msec
glassBreakRampDevWithParameters_v3	TanhBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v2_16000	108.83sec	TPR = 22 / 23	FPs = 5	52.4msec
glassBreakRampDevWithParameters_v3	TanhBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_Short_v1_16000	26.81sec	TPR = 4 / 5	FPs = 0	89.8msec

129/148 59

Sophisticated timing constraints, i.e., shatter after thud, thud coincides with high frequency impulse, suppress thud after shatter

Log

Code Version	Chain	Wav File	Duration	TPR	FPs	Latency
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v1_16000	151.72sec	TPR = 20 / 20	FPs = 11	81.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v1_16000_mixed_included	247.02sec	TPR = 50 / 50	FPs = 19	58.1msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_Training_v1_16000	597.46sec	TPR = 46 / 50	FPs = 33	110msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_v2_16000	108.83sec	TPR = 21 / 23	FPs = 4	62.3msec
glassBreakRampDevWithParameters_v3	LogBaselineAndZCR_hfPeaks_v3Timing	GB_TestClip_Short_v1_16000	26.81sec	TPR = 4 / 5	FPs = 0	76.3msec

141/148 67