

## Predobdelava stereo zvoka

Pred izvedbo kompresije MDCT stereo zvočni signal (levi L in desni R kanal) razdelite na kanala M in S (po pristopu Mid-Side), kjer veljata sledeči enačbi:

$$M = (L + R) / 2$$

$$S = (L - R) / 2$$

Nato nad M in S posebej izvedite MDCT transformacijo ter binarni zapis v datoteko kot je opisano v nadaljevanju. Pri dekompresiji lahko dobite original signala L in R (po izvedbi IMDCT), tako da uporabite sledeči enačbi:

$$L = M + S$$

$$R = M - S$$

## Kompresija zvoka s pomočjo MDCT

Modificirana diskretna kosinusna transformacija (MDCT) je podobna DCT-ju, le da se transformirajo prekrivajoči se bloki velikosti  $2N$ , kjer prva polovica nekega bloka sovpada z drugo polovico prejšnjega bloka. Algoritem je zelo učinkovit za kompresijo zvoka, zato se uporablja v znanih formatih kot so npr. MP3, AC-3, Vorbis in AAC. Pred izvedbo transformacije je potrebno signal razdeliti na enako velike bloke določene velikosti. Prav tako je potrebno dodati  $N$  ali več ničel na začetku in koncu signala (prvi in zadnji blok).

Pred izvedbo transformacije pretvorite še bločni signal  $x$  s pomočjo okenske funkcije katera bo zgladila frekvence ob robu bloka za nižanje napake pri transformaciji neperiodičnih signalov. Pri tem uporabite okensko funkcijo:

$$w_n = \sin \left[ \frac{\pi}{2N} \left( n + \frac{1}{2} \right) \right]$$

Nato z zgornjo enačbo ustvarite signal okenske funkcije velikosti danega bloka ( $2N$ ) in pomnožite vzorce z enakoležnimi vzorci okenske funkcije (npr.  $x_n = x_n * w_n$ , pri čemer  $n \in [0, 2N]$ ). Nato sledi 1D MDCT transformacija bloka  $x$  v izhodni blok  $X$  na sledeč način:

$$X_k = \sum_{n=0}^{2N-1} x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} + \frac{N}{2} \right) \left( k + \frac{1}{2} \right) \right]$$

, kjer  $0 \leq k < N$ ,  $0 \leq n < 2N$ . Velikost vhodnega bloka  $x$  je  $2N$ , izhodnega bloka  $X$  pa  $N$ . Število izhodnih koeficientov na posamezni blok je polovico manj kot pri vhodnem bloku.

Po transformaciji koeficiente najprej zaokrožite v cela števila. Nato v vsakem bloku zadnjih  $M \leq N$  koeficientov nastavite na 0.  $M$  določa stopnjo stiskanja.

Kompresiran signal shranite v binarno datoteko. Na začetku datoteke generirate glavo datoteke, kjer shranite število vseh vzorcev (4 zlogi), N (2 zloga), frekvenco vzorčenja (4 zlogi) ter stopnjo stiskanja M (2 zloga). Nato hranite posamezne MDCT bloke velikosti (N - M), kjer vsak koeficient hranite z naslednjo shemo:

dolžina za koeficient      koeficient  
6 bitov                              dolžina bitov

Pri dekompresiji najprej izvedete inverzni MDCT (IMDCT). Pri tem transformiran signal X pretvorite v (morebitno izgubni) signal y:

$$y_n = \frac{2}{N} \sum_{k=0}^{N-1} X_k \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} + \frac{N}{2} \right) \left( k + \frac{1}{2} \right) \right]$$

, kjer  $0 \leq k < N$ ,  $0 \leq n < 2N$ .

Po IMDCT ne pozabite na množenje y z okensko funkcijo. Rekonstrukcija original signala x se nato izvede s seštevanjem prekrivajočih izhodnih koeficientov med dvema sosednjima blokoma, kot je prikazano v primeru. Izhodne vrednosti zaokrožite navzgor.

### Primer transformacije MDCT in IMDCT (brez kvantizacije)

Vhod iz 12 vzorcev:

1	2	3	4	5	6	7	8	9	10	11	12
---	---	---	---	---	---	---	---	---	----	----	----

1. Razdelitev na bloke velikosti 6 (N = 3):

0	1	4	7	10
0	2	5	8	11
0	3	6	9	12
1	4	7	10	0
2	5	8	11	0
3	6	9	12	0

Bloki se prekrivajo čez polovico. Prav tako so na začetku in koncu signala dodane ničle, kjer ni podatkov.

2. Aplikacija okna nad vsakim blokom:

0	0.2588	1.0353	1.8117	2.5882
0	1.4142	3.5355	5.6569	7.7782
0	2.8978	5.7956	8.6933	11.5911
0.9659	3.8637	6.7615	9.6593	0
1.4142	3.5355	5.6569	7.7782	0
0.7765	1.5529	2.3294	3.1058	0

3. Izvedba MDCT nad vsakim blokom:

-2.9232	-9.5459	-15.9099	-22.2739	-4.5015
-1.4142	-0.8966	-0.8966	-0.8966	9.0029
0.0947	-0.1641	-0.1641	-0.1641	-4.5015

**Inverz:**

1. Izvedba IMDCT:

0.0000	-2.6390	-4.7603	-6.8816	-9.0029
0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	2.6390	4.7603	6.8816	9.0029
1.7424	5.4166	9.0909	12.7651	0.0000
2.8284	7.0711	11.3137	15.5563	0.0000
1.7424	5.4166	9.0909	12.7651	0.0000

2. Aplikacija okna nad vsakim blokom:

0.0000	-0.6830	-1.2321	-1.7811	-2.3301
0.0000	0.0000	0.0000	0.0000	0.0000
0.0000	2.5490	4.5981	6.6471	8.6962
1.6830	5.2321	8.7811	12.3301	0.0000
2.0000	5.0000	8.0000	11.0000	0.0000
0.4510	1.4019	2.3529	3.3038	0.0000

3. Rekonstrukcija original signala s seštevanjem:

-0.0082	-0.6830	-1.2321	-1.7811	-2.3301
0.0000	0.0000	0.0000	0.0000	-0.0000
0.0305	2.5490	4.5981	6.6471	8.6962
1.6830	5.2321	8.7811	12.3301	0.0000
2.0000	5.0000	8.0000	11.0000	0.0000
0.4510	1.4019	2.3529	3.3038	0.0000

Izhod:

$$1.6830 + (-0.6830) = 1$$

$$2.0000 + 0.0000 = 2$$

$$0.4510 + 2.5490 = 3$$

$$5.2321 + (-1.2321) = 4$$

1	2	3	4	5	6	7	8	9	10	11	12
---	---	---	---	---	---	---	---	---	----	----	----