I. Abstract and Motivliation:

In this problem, we referred to two papers to do the mode skip to save the encoding time without decreasing the R-D cost performance that much. The two techniques are fast inter mode decision and fast intra mode decision respectively. Exhaustive mode search can get the best RD cost but it may take a lot of time. These two papers find methods to skip some mode estimation according to the property of the video, such as the property of the homogeneous region and the dependency of the neighboring regions.

In this problem, I followed the instruction of the Homework and the papers to implement the two algorithms. But I use the best inter SSE to compare with the Threshold instead of SAD in part2, the reason is explained in the discussion part.

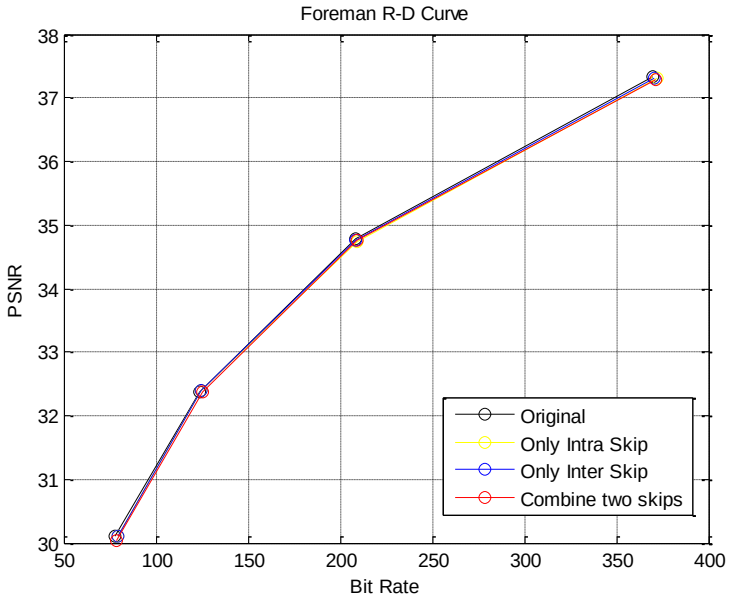
II. Approach and Procedure:

In the fast full search mode, for each 16x16 Macroblock, the encoder does the motion search within the search range (the search pivot is decided by the prediction of the motion vector). Then it stores the motion vector who has the smallest cost for the block. In this step, the 4x4 block is for storage but not for independent motion vector(in the *currMB->p_SetupFastFullPelSearch()*),this is the meaning of fast full search: storage).For a specific partition type, the encoder find the best motion vector for each subblock based on best SAD cost. Finally, the encoder select the best inter or intra mode in a loop to compare the different modes' RDcost.
.

The motion vector search is first for the 16x16,16x8, 8x16 types. To implement the [1]'s technique, we can set a "*if*" judgement just after the 3 first motion vector searches to disable or enable the further finer partition. When we read a new frame from the file, we first calculate the MAFD between the new frame and the stored previous frame, and calculate the MAD of each MB in the frame. Then we save the current frame to the buffer (the previous frame). We can set a global array storing the skip information when we encode one Macroblock.

The intra mode is compared with the inter modes together in the loop containing compute_mode_RD_cost function after all inter modes' motion vectors has decided. I use a global array to save the best RDcost of the current MB to use later. After comparing between the inter modes, I can get the SSE for best inter modes to judge whether skip the following intra modes or not.

III. Result:


Foreman R-D Curve
PSNR
Bit Rate
38
37
36
35
34
33
32
31
30
50
100
150
200
250
300
350
400
Original
Only Intra Skip
Only Inter Skip
Combine two skips

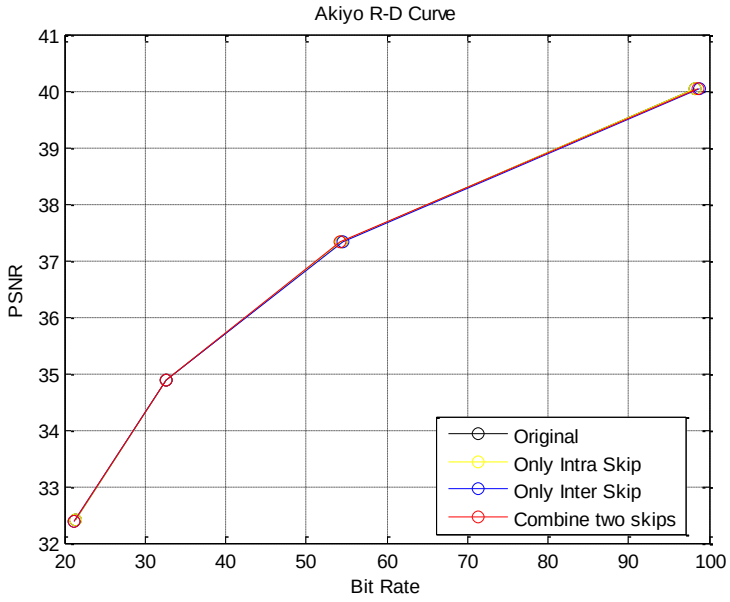

Akiyo R-D Curve
PSNR
Bit Rate
41
40
39
38
37
36
35
34
33
32
20
30
40
50
60
70
80
90
100
Original
Only Intra Skip
Only Inter Skip
Combine two skips

Table 1. the result of Akiyo

QP	PSNR(Y)	Bit Rate	ME time	Total Time	inter skip	intra skip	Save Percenterge
28	40.037	98.19	136.436	145.576	off	off	*
32	37.338	54.42	138.066	146.995	off	off	*
36	34.893	32.65	140.09	148.873	off	off	*
40	32.403	21.33	142.425	151.096	off	off	*
28	40.038	98.24	114.347	122.591	on	off	15.21%
32	37.325	54.33	113.613	121.469	on	off	17.36%
36	34.893	32.64	112.838	120.434	on	off	19.10%
40	32.403	21.33	110.639	117.981	on	off	21.92%
28	40.043	98.88	136.72	144.834	off	on	0.51%
32	37.34	54.55	138.709	146.726	off	on	0.18%
36	34.887	32.58	139.996	147.664	off	on	0.81%
40	32.384	21.21	142.763	150.425	off	on	0.44%
28	40.041	98.69	115.03	122.146	on	on	16.09%
32	37.329	54.24	114.539	121.447	on	on	17.38%
36	34.888	32.57	113.569	120.154	on	on	23.90%
40	32.384	21.21	111.861	118.306	on	on	21.70%

Table2. the result of Foreman

QP	PSNR(Y)	Bit Rate	ME	Total	Inter skip	intra skip	Save Percenterge
28	37.329	369.71	139.146	149.104	off	off	*
32	34.785	208.43	142.162	151.515	off	off	*
36	32.381	123.79	144.767	153.72	off	off	*
40	30.094	78.07	146.988	155.749	off	off	*
28	37.31	371.81	117.121	126.082	on	off	15.44%
32	34.746	208.99	115.845	124.076	on	off	18.10%
36	32.391	124.28	115.296	123.145	on	off	18.72%
40	30.108	79.18	110.77	118.071	on	off	24.19%
28	37.309	370.19	139.37	147.92	off	on	0.79%
32	34.766	208.36	141.824	149.996	off	on	1.00%
36	32.398	124.7	144.555	152.424	off	on	0.84%
40	30.106	79.23	147.186	154.924	off	on	0.53%
28	37.294	371.73	117.882	125.485	on	on	15.84%
32	34.756	209.49	116.992	124.158	on	on	18.05%
36	32.378	125.38	114.916	121.663	on	on	20.85%
40	30.044	78.42	111.913	118.325	on	on	24.75%

IV. Discussion:

(1) For the first technique, the fast inter mode decision, we compare the MAD of the Macroblock (the current one and the collated one in the previous frame) and the MAFD of the frames(the current one and the previous one). If MAD.w≤MAFD holds, we can skip the inter modes for further partition equal or less than 8x8.

If the Macroblock is in homogeneous region, it is enough to use large block size to do motion estimation. if the Macroblock includes many objects which are moving to different directions or the Macroblock is on the boundary of a moving object, it is better to partition the Macroblock into smaller size to consider independently for each motion vector. The MAD between MBs in two neighboring frames can reflect the change level of the Macroblock. If the condition "≤" mentioned above holds, it implies that this Macroblock is homogenous along time compared to the whole frame. Then we need not partition the current Macrobolck into further small sizes. The 16x16, 16x8, 8x16 partition mode is enough for motion estimation. Then we skip other inter modes. We applies a coefficient w based on Qp in the judging condition. The reason is smaller Qp will often result in smaller block types. Thus the w is bigger to make the condition more strict to satisfy to avoid terminate the mode search for small Qp too early.

(2) For the second technique, the fast intra mode decision. The rationale is that the neighboring Macroblocks are often high correlated. In this way, we can use the encoded neighboring MB's RD cost to approximate the current encoding MB's RD cost.
We use minimum vaule of the 4 neighbors ' RD cost to be a threshold to approximate the current MB's RD cost.

$$T_{kl} = \min\{RDcost(m,n)|(m,n) \in Neighbors\} \approx RD(Current\ MB) \geq SSE(Current\ MB)$$

If we use inter mode, we should satisfy that the T_{kl} is larger than the SSE of the current best inter mode. In the code, the RDcost of inter mode is calculated earlier than that of intra mode. If the " $\geq$ " cannot hold for the best inter mode, it means that the inter mode is not good enough since the distortion is too much mainly due to the low correlation in different frames. We should still find if there exist good intra modes. Or we can think the best inter mode satisfies the condition and skip all the intra modes.

(3) The RD cost of 4 modes for 2 bitstreams are shown above. We can find not much quality decreasing after using the 3 skip modes. But each mode had saved time compared to the original encoding. And the combination of the two methods has saved extra time

compared to both skip techniques. This result makes sense. It proves that these two methods have saved time with a little change of RD cost.

The fast inter mode skip saved a lot of time (about 20%, see the table) without obvious RD cost quality decreasing. It proves this method is meaningful. For a specific technique, the Large Qp seems to save more time, because the condition of inter mode skip for samller Qp is more strict for satisfying with larger w.

I have expected the saving time (relative percentage based on the original encoding) of Foreman should be less than that of Akiyo. Because Akiyo is almost the same in the two neighboring frames and we should not partition it into smaller blocks. But the result I got did not conform to my expectation strictly. One reason may be the unstatblity of computer system since the values are not differs so much. The original Foreman will not need encode so much small partitions, either. Actually, for Akiyo, the background seldom moves, the FMAD is also small. It is also hard to satisfy the skip condition considering this.

For intra mode, we can see the saving time is very small. I have modified the method (have asked the TA) from [2], to compare the SSE rather than SAD of best inter mode. This is more reasonable, because the [2] calculate RD cost using SAD, but the JM coding under our configuration is using SSE. We should make the two things comparable, thus it is reasonable to modify the SAD to SSE.

The [2] got good result. I think the difference may exist in the mode selected. The [2] mentions that under high-profile the intra mode will take a lot of portion in the whole ME. But we are in the Profile (level=77) which is the main profile. Even if we skip all the intra modes directly without condition, the saving time is still limited. Another reason is we use reference frame number 5 rather than 1[2], which will lead the time saving is not so much obvious. The time saving of Akiyo and Foreman are both small and differs little. But it seems that the Foreman saves a little more. It is because that the Foreman are mainly depends on inter coding. It is much easier to skip the intra modes.

V. References:

[1] Jing, X., and L-P. Chau. "Fast approach for H. 264 inter mode decision." *Electronics letters* 40.17 (2004): 1050-1052.

[2] Kim, Byung-Gyu. "Fast selective intra-mode search algorithm based on adaptive thresholding scheme for H. 264/AVC encoding." *Circuits and Systems for Video Technology, IEEE Transactions on* 18.1 (2008): 127-133.

[3] Lecture Notes
[4]Discussion Slides
[5]Homework Introduction