

Q1: Consider a 16x16 binary image with the same number of black and white pixels inside (128 for each). What block pattern will make JBIG-1 encoder perform best? What will be the worst?

Ans:

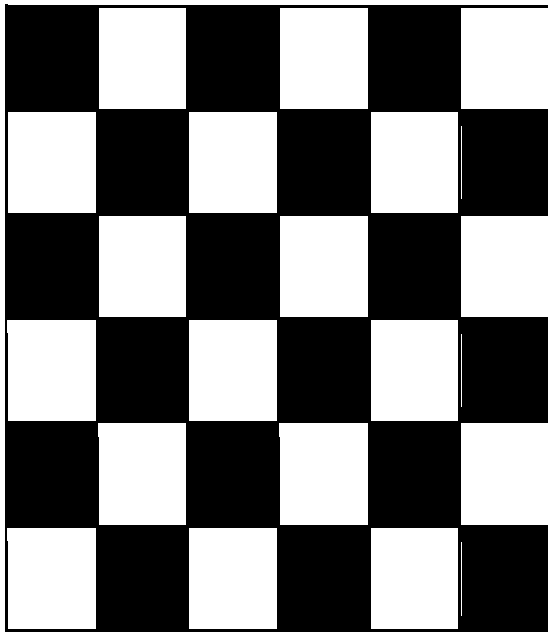
1. Best case:

When black pixels are clustering at top/down area, because it can be reduced to 2 pixels with Typical Prediction only.

			
...
...
			

2. Worst case:

The worst case is when only can be compressed by Adaptive Arithmetic Coder. The following chessboard like case should be the worst case, because as for Typical Prediction will not have any usage at this case, as for Deterministic Prediction, there is no any block can be predicted (checked the table), and for model template or adaptive template the entropy should be the same for every pixels (use default model), so only Adaptive Arithmetic Coder can solve this kind of problem. And also after resolution reduction we will get the same pattern but just a quarter size one.



Q2: In the worse case, can you design a customized RR/DP/AT to improve the coding performance?

Ans:

1. I think we can redesign Adaptive Template to resolve this kind of problem.
 Let's use horizontal-reversed "L" shape Template when the up and left pixels are the same color
 then the current pixel is the other one (only care high solution).
 For example, assume foreground is 1 and background is 0, let up/left be 1 and if after the convolution with the template is 2 then the current is background, and if the result is 0 then the current
 In this way the model size is smaller and almost all pixels can be predicted by the template (except first row).

EX:

