In current existing coding standards, three basic picture (slice) coding types are designed depending on the degree of freedom for generating the prediction signal: I frame (intra prediction coded frame a.k.a. intra frame), P frame (forward prediction coded frame), and B frame (bi-directional prediction coded frame). For I frame, intra-prediction is utilized to remove the correlation between image block and its adjacent pixels, without any dependence on other frames, therefore, I frame has the feature of independent encoding and decoding, which enables I frame hold ‘special’ position in video coding. I frame is the “starting point” of a video sequence, while the following P frames or B frames directly or indirectly take I frame as reference frame. Besides that, I frame has the functionalities of random access that allows stream switching and error resilience that prevents spatial and temporal spread of transmission error. In some special applications such as video editing, video streaming, and high-definition movie compression, I frame appears in a very high frequency, affording the main roles of compression tasks. Considering all the above, H.264/AVC has defined four all-I-frame profiles [1] which include High 10 Intra Profile, High 4:2:2 Intra Profile, High 4:4:4 Intra Profile, and CAVLC 4:4:4 Intra Profile. In the subsequent H.264 SVC which is the scalable extension of H.264/AVC has also defined an all I-frame profile [2]: Scalable High Intra Profile.

Intra predictions in MPEG4 H.264/AVC perform zero-order extrapolation using the adjacent boundary samples of reconstructed neighboring blocks to get the prediction of the entire coding block., and the adjacent boundary samples may be the neighboring row of samples on top of current block (A–H), or the neighboring column of samples on the left (I–L), or the both (A–L plus Q) for 4x4 block. Nine kinds of different intra prediction modes are designed to adapt to different image texture features. The nine modes consist of eight directional modes and a DC prediction mode, as shown in Figure 1. In video streaming, intra-coded frame or block is also widely employed to suppress temporal propagation of transmission error due to packet loss, and in these application circumstances, intra coding (frame or block) is forcibly determined by some special rules, so this intra coding is known as the alias of intra refresh. Comparing with inter-frame prediction coding, the residual produced by intra prediction is still very large, and the average bits per I frame are usually 4 times more than those of P frame. Intra coding faces great requirements to further improve its coding performance.

Through perspective analysis, it can be found that there are three significant limitations in intra coding. First of all, limited modes are insufficient to cover various texture features inside image content, even though H.264/AVC contains 9 prediction modes. Secondly, prediction asymmetry, which means neighboring boundary top row of samples or left column of samples or both of them are still 1-D array, but the destination block to be predicted is 2-D array, resulting prediction accuracy is not guaranteed especially when the samples in current block are far from boundary samples. In the end, temporal prediction can not be allowed in intra coding, with the purpose of permitting independent decoding functionality, which is widely deployed in error-resilient video streaming, random access and stream switching.

After the standardization of H.264/AVC, great efforts
have been taken to improve intra coding’s performance in industry and academic. On summary, they can be classified into two main categories according to original motivations. The first category is aiming to explicitly enhance prediction accuracy of intra coding, which is termed as ‘enhancing intra prediction’ afterwards, and the other one is trying to modify the following coding components (such as transform, scan etc.) with the purpose of making them more suitable for intra prediction and then improving the whole performance of intra coding, and this kind of basic thought is termed as ‘enhancing following components’ afterwards. In the category of enhancing intra prediction, there are four sub-classes: (1) **Distant prediction.** This type includes famous. template matching prediction proposed by T.K.Tan et al. [3] and intra displacement compensation proposed by Yu et al. [4], both of which lie in searching for better matching blocks in the already reconstructed distant image part, giving up the restriction of adjacent boundary pixels in conventional intra coding. (2) **More modes or combination of existing modes.** More prediction modes can cover more texture details, and then can bring stable coding gain [5]. Yan Ye et al. [6] tried to introduce more prediction modes by combining existing modes, and 3% bit-rate saving was achieved. Recently, some modifications to the traditional intra coding in H.264/AVC are also proposed in VCEG group, which includes geometry adaptive bi-partitioning for prediction [7], adaptive multi-directional intra prediction [8], bi-directional intra prediction [24], and all of them can also be classified to this sub-class. (3) **Intra adaptive filtering prediction.** Inspired by adaptive interpolation filtering in inter prediction, Limin Liu et al. [9] adopted least squared error principle to obtain current block’s best prediction from neighboring pixels, but this method selected training window in adjacent frames, and this effort impaired the independence of I frame and only had theoretic contributions. (4) **Inpainting based intra prediction.** This work [10] extracted assistant information to guide inpainting process to generate the prediction of current block, and brought significant improvement on intra coding with single block partition, but the gain was not obvious for variable block sizes. In the category of enhancing following components, there are also several sub-classes: (1) **K-L transform.** According to the different characteristics of residuals after intra prediction, the best transform basis (K-L transform basis) was trained by Ye Yan et al. [6], and this technique was also integrated into KTA software as mode-dependent directional transform (MDDT). The same idea was also reported in intra predictive transform (IPT) [11], which proved that the combination of Wiener filtering prediction and K-L transform could reach the best performance in a strict mathematical form. (2) **Directional filtering transform** [12]. In this type of technique, transform module was connected to intra prediction in a tight way, and there was even no direct DCT transform after intra prediction, while the redundancy inside block was removed in an elaborate filtering way. (3) **Residual coefficient re-arrangement.** This thought was first implemented to make the residual matrix having maximum number of zeros after DCT transform [13], and adaptive scanning method for intra coding [14] is also belonging to this sub-class.

Though various methods have been proposed to improve the performance of intra coding, significant coding gain is not achieved, and their potentials seem still limited and not enough for the standardization of next-generation high performance video coding standard (HVC or NGVC). To break through such bottlenecks, under the premise of satisfying intra block or intra frame’s basic functionalities, weaving some prediction constraints in some special way may be an effective taste.

As a matter of fact, the features of intra-coded frame or block would not be impaired given the same faithful reference in the encoder and decoder, and in another word, under specific conditions intra frame or block can also be temporally predicted implicitly just like [15]. In this paper, a new intra coding and refresh method based on video epitomic analysis is proposed. Image epitome generated by epitomic analysis is considered as image’s most essential representation, which preserves the global texture and shape characteristics in image. The size of image epitome is only a fraction of that of original image or even less. After the introduction of image epitome, the globally variation information in image can be taken as prior knowledge (image epitome a.k.a epitomic priors), and can be applied to generate prediction. The new intra prediction method presented in this paper can not only eliminate the correlation between current block and the overall image, but also can reduce the temporal redundancy among adjacent images to some extent while avoiding dependency of current image on other coded images. There are two main contributions in this paper: First, this paper is the first time to introduce image epitome into the hybrid video coding framework for enhancing the accuracy of predictive coding, and different prediction strategies are designed to approach the best rate-distortion performance. Second, to get a good image epitome for compression task, compression-oriented video epitomic analysis is proposed with the purpose of getting image epitome suitable for compression task. This paper is organized as follows: in the second section, intra coding framework with epitomic priors is presented. Video epitomic analysis will be described and compression-oriented video epitomic analysis is formulated in the third section. The fourth section will focus on the new intra coding methods based on image epitome. The experimental results and analysis will be reported in the fifth section. We will conclude in the last section.

II. INTRA CODING FRAMEWORK WITH EPITOMIC PRIORS

The proposed intra coding framework with epitomic priors is illustrated in Figure 2, where the left dash-line box is the proposed encoder and the right is the proposed decoder. Different from knowledge based coding (KBC) [16], the priors are extracted, coded and transmitted in an online way, but the KBC technique requires the same knowledge database to ensure consistent decoding results in both encoder and decoder, and the knowledge is not coded and transmitted. For this new coding framework, in terms of compression, the best coding efficiency can be achieved by minimizing the joint rate-distortion cost, in which the rate should include the overhead for epitomic priors.
From the perspective of KBC, the priors are defined as image representation after factoring large-scale repeated image content. How to obtain such kind of energy-compacted coding priors is not a trivial work. The epitomic analysis on image or video which is first reported by N. Jojic and Vincent Cheung [17, 18] has given us inspiration on how to extract the coding priors. Image epitome which is calculated after epitomic analysis is considered as image’s most essential representation, which preserves the global texture and shape characteristics in image, and the size of image epitome is only a fraction of that of original image or even less by fusing similar textures. So image epitome is very suitable for being considered as coding priors. Conventional epitomic analysis gains popularity in accomplishing image processing tasks such as image denoising, image inpainting, image super-resolution etc. However, compression has its own characteristics, and to maximize the coding efficiency, the first key component is to compute the epitomic priors according to the requirements of compression purpose. This process will be discussed in the following section in detail.

Besides extracting epitomic priors suitable for compression, another key problem is how to generate prediction under the condition of given epitomic priors. Taking signaling overhead into account, two new prediction modes are designed to generate prediction from epitomic priors inspired by template matching prediction and intra displacement compensation. The two new modes are combined with existing prediction modes to form new candidate mode set, and the best prediction mode is selected from the set by rate-distortion optimization (RDO).

Epitomic priors can be extracted from single frame and used in its coding, and also can come from several frames, and can be applied as epitomic priors for those frames’ coding. In the case of several frames sharing the same image epitome, the temporal relationship can be constructed through the intermediate bridge of image epitome in a hidden way for these frames. If this kind of priors is applied in intra coding, temporal redundancy among adjacent images can be removed to some extent while avoiding the dependency of current image on other coded images.

If only one frame is fed into video epitomic analysis, its corresponding image epitome is generated, and when this image epitome has good enough visual quality, and it can be used as the base-layer image in spatially scalable video coding while original image is taken as enhancement layer image. This re-sizing process to obtain base-layer video is named as epitomic resizing. Epitomic resizing may be advantageous over down-sampling method and cropping method which are used in current scalable extension of H.264/AVC to get base-layer video.

Considering the peak change in image content between different video shots, it’s better to take epitomic analysis on each video shot to avoid information fusing of distinctively different content in different shots, and each video shot has its own epitomic priors. Video shot can be detected by several classical methods, which are beyond the scope of this paper and in-depth discussions can be referred to [19]. The priors should be coded in a lossless manner and transmitted to decoder side. In addition to the usage of prediction, epitomic priors can be also be used to reconstruct the lost video slices for error concealment during transmission.

Video epitomic analysis requires buffering multiple frames, which would cause time delay, so it’s not suitable for real-time applications. But this problem can be alleviated by some special ways. It’s observed that the images in the same video shot has great similarity, so to simplify computation process of image epitome and satisfy the real-time requirements, only the first image is fed into
video epitomic analysis process, and the image epitome is considered for the image epitome for the following video frames until the next video shot is detected. In this case, if the size of image epitome is equal to that of single frame, the epitomic priors are becoming the first frame in video shot, and the coding strategy evolves into inter-I-frame prediction with the first frame always as prediction source. But, the size of image epitome should not be large, because it would bring more overhead of coding image epitome and more computational complexity for generating prediction, and image epitome should be more essential and concise. Therefore, better coding performance can be expected comparing with inter-I-frame prediction (taking only part of the first frame as reference to satisfying size requirement), as image epitome can cover more video content.

III. VIDEO EPITOMIC ANALYSIS

3.1. N.Jojic’s video epitomic analysis

Image epitome preserves enough texture and shape characteristics in image. The size of image epitome is only a fraction of that of original image or even less. Given an image epitome and its original image(s), a mapping between them can be derived, and if the mapping and image epitome are given, the original image can also be reconstructed. But in real practices, both the mapping and image epitome are unknown, and the problem can be formulated as Maximum Likelihood Estimation (MLE) problem [17] with EM (Expectation Maximization) algorithm can be applied to solve this problem. The method in [18] divides the original image into a set of image patches with different size, and the image patches can overlap with each other, and the detailed process is in the following:

\[
\phi(x,z) = \frac{1}{\left| X^* \right|} \sum_{p \in X^*} \left| x_p - z_{p'} \right|^2
\]

\[\text{Image 52x248 to 196x365}\]

In the above equation, \( x \) represents the original image, \( z \) for image epitome, \( X^* \) denotes image patch set sampled from original image, and the variable \( p \) represents the coordinate set of image patch in \( X^* \), and \( p^* \) stands for the coordinate set of best match in image epitome. \( x_p \) is the pixel value vector in the coordinates of \( p \) while \( z_{p'} \) is the vector composed of pixel values in the coordinates of \( p^* \). Therefore, the problem can be formulated as a minimization problem:

\[ z_{best} = \arg \min \phi(x,z) \]

which can be solved by EM algorithm with multiple iterations. Before EM algorithm, \( z \) should be initialized randomly. During each iteration, E-step and M-step would be done alternatively until termination conditions are met:

E-step: Searching the best matching block \( z_{p'} \) for every \( x_p \) in \( X^* \), as illustrated in Figure.3;

M-step: Updating \( z \) for minimizing the energy function \( \phi(x,z) \) according the sets of \( z_{p'} \) and \( x_p \), the detailed updating way is shown in equation (3):

\[ z(i, j) = ( \sum_{p \in X^*} \delta_{p^*} (i, j))^{-1}( \sum_{p \in X^*} z_{p^*}(i, j)) \]

In the above equation (3), \( p \) and \( p^* \) are the matching coordinate set pair through E-step, and \( z_{p^*}(i, j) \) represents the pixel values of the corresponding pixel in \( X^* \) with the location of \( (i, j) \) in \( p^* \). The function \( \delta_{p^*} (i, j) \) is the delta function which is described by equation (4):

\[ \delta_{p^*} (i, j) = \begin{cases} 1 & \text{if } (i, j) \in p^* \\
0 & \text{if } (i, j) \notin p^* \end{cases} \]

After the updating process, current iteration is finished, and E-step in the next iteration is started.

3.2. Compression-oriented video epitomic analysis

From the perspective of residual’s energy, minimizing the target function \( \phi(x,z) \) shown by equation (1) can approximate the best coding performance, however, the coding results shown in Figure 13 are not as what we expect, indicating there are more factors influencing coding efficiency except minimizing \( \phi(x,z) \). The epitome obtained from section 3.1 is not suitable for compression purpose due to four reasons: first of all, some blocks can be predicted very well by conventional directional prediction, for example, smooth image regions which is shown in Figure 5., and edge regions usually can not be predicted accurately because edge direction can not always match predefined prediction direction; secondly, the averaging strategy in M-step illustrated by equation (3) makes image epitome blurry, restricting the prediction accuracy from image epitome. The third one is that, to fuse similar texture information as more as possible, the \( x_p \) is sampled from original image in an overlapped way, and it is different from block partition in compression. The last one is the lack of coherence in image epitome, causing the overhead of coding image epitome and the mapping between original image and image epitome cost too much.

Figure 3. Image epitome (the right) and the mapping between original image (the left) and image epitome
A new energy function considering all above compression requirements is formulated as follows:

$$\phi(x,z) = \frac{1}{|X|} \sum_{p \in X} \delta_p \left| x_p - z_p \right|^2 + \frac{\alpha}{|Z^+|} \sum_{q \in Z^+} \left| x_q - z_q \right|^2$$

(5)

**Adaptive weighting on completeness term**

It is expected that the new intra prediction method and existing intra prediction methods can be complementary to each other. For the regions where cannot be predicted well by existing intra prediction modes, their information should have much higher probability to appear in image epitome, and in another words, they should provide more contributions to image epitome more than those regions where can be predicted well by existing prediction modes. Therefore, adaptive weighting on completeness term is introduced for each patch \( x_p \), whose contribution to the energy function \( \phi(x,z) \) is \( \delta_p \left| x_p - z_p \right|^2 \). The weighting parameter \( \delta_p \) can be measured by residual energy of patch \( x_p \), but such operation needs two-pass coding in which the first pass is to get residual after intra prediction, as shown in Figure 5. However, the two-pass process costs great computational complexity. It can be observed that the residual energy is large in regions which contain rich edges, so it’s very convenient to use the edge intensity (just like the right image in Figure 4.) to estimate residual energy. In our experiments, classical Laplace operator is used to extract edge map.

**Coherence constraint**

Figure 5. Residual after intra prediction (the left) and edge map using Laplace operator

Video epitomic analysis in section 3.1 only take image’s completeness into account, and the inferred image epitome shows great artifacts due to the lack of coherence, and for explicit coding of the mapping relationship between original image and image epitome, it would cost much overhead due to the poor correlation between adjacent patches. In the new energy function (5), the coherence term \( \frac{\alpha}{|Z^+|} \sum_{q \in Z^+} \left| x_q - z_q \right|^2 \) is added, and in this term \( Z^+ \) denotes image patch set sampled from the image epitome, and \( z_q \) represents the image patch on the coordinate set \( q \), and \( x_q^* \) stands for best match for \( z_q \) in the original image on the coordinate set \( q^* \). \( \alpha \) is the modulation factor, and in our experiments, \( \alpha \) is set to 0.03. After introducing coherence constraint, the energy function (5) seems very similar to the energy function based on bidirectional similarity proposed by Denis Simakov et al. [20], which was used to do editing tasks such as image resizing, image re-shuffling etc., but every term of target energy function in [20] has different meanings. The size of patch is set equal to that of block in video compression, and each patch is not overlapped with each other in completeness term. In coherence term, overlapped patches are sampled as same with that in section 3.1

**Robustness to outliers**

EM algorithm can be also applied to minimize the target energy function \( \phi(x,z) \) represented by equation (5):

--E-step: Searching the best matching block \( z_p^* \) for every \( x_p \) in \( X^+ \), and then searching the best matching block \( x_q^* \) for every \( z_q \) in \( Z^+ \);

--M-step: Updating \( z \) with the objective to minimize the energy function \( \phi(x,z) \) according to the searched sets of \( x_q^* \) and \( x_p^* \), as not all patches are equally reliable, and the averaging of all pixels pointing to current pixel in image epitome will cause blurry. If there were outliers, the convergence would become hard. So, the most likely pixel value at location \( (i, j) \) will minimize

$$\sum_{\alpha} \delta_p \left| w_{p,n}(z(i,j) - x_{p,n}(i,j)) \right|^2 + \alpha \sum_{q} w_{q,n}(z(i,j) - x_{q,n}(i,j))^2$$

, therefore:
The weights defined by equation (7) make sure the samples which are more similar to current sample would have more contribution to the updated current sample.

\[
\begin{align*}
    z(i,j) &= \left( \frac{1}{|X|} \sum_{p \in X} \delta_p \cdot w_p(i,j) + \frac{\alpha}{|Z|} \sum_{p \in Z} w_p(i,j) \right)^{-1} \\
    &= \frac{1}{|X|} \sum_{p \in X} \delta_p \cdot w_p(i,j) \cdot x_p(i,j) + \frac{\alpha}{|Z|} \sum_{p \in Z} w_p(i,j) \cdot x_p(i,j)
\end{align*}
\]

(6)

\[
\begin{align*}
    w_p(i,j) &= e^{-f_1(i,j) - z_i(i,j)} \\
    w_z(i,j) &= e^{-f_2(i,j) - z_i(i,j)}
\end{align*}
\]

(7)

The weights defined by equation (7) make sure the samples which are more similar to current sample would have more contribution to the updated current sample.

3.3 Epitomic resizing for spatial scalability

In the case of one frame with one image epitome, if image epitome is visually good enough, it can be taken as base-layer image in spatially scalable video coding, coding efficiency of which can be improved as the base-layer image is derived to maximize coding efficiency of the enhancement layer image. Therefore, the visual quality of image epitome is as important as the characteristics for coding performance for this application. To make good trade-off between coding performance and visual quality, the adaptive weighting on completeness term is not integrated to ensure the ability to preserve all information appearing original image. At the mean time, the limited space (size) resource requires similar texture to merge as much as possible, and this will bring the loss of significant contours and edges in image, just as shown in Figure 6(c), and visual quality would be degraded greatly. To solve such problem, the detail information can be extracted and used to boost patch search to get more accurate results which preserve much more detail information.
Taking one-dimensional signal for example (illustrated by Figure 7.), there are two candidates for $f(x)$ in search process: $g_1(x)$ and $g_2(x)$, the color difference of which are equal: $\int (f(x) - g_1(x))^2 dx = \int (f(x) - g_2(x))^2 dx$. But $g_2(x)$ should be preferred, as it contains more detail similar to that of $f(x)$ than $g_1(x)$.

To avoid the quality degradation, detail information can be taken as detail channel, which is added to YUV channel to highlight detail content in original image. Detail channel is extracted through bilateral filtering [26] by equation (9):

$$\hat{x}_i = \frac{1}{K_i} \sum_{j=1}^{\Omega} f_i \| \| x_j - x_i \| g(x_j) \| x_j - x_i \|$$

in which,

$$K_i = \sum_{j=1}^{\Omega} f_i \| \| x_j - x_i \| g(x_j) \| x_j - x_i \|$$

$f$ and $g$ are spatial Gaussian filter centering location $i$ and sample value Gaussian filter centering $x_i$, and $K_i$ represents normalization factor, and $\Omega$ is the neighboring domain for bilateral filtering. Detail channel is got by $x_i \rightarrow \hat{x}_i$. The combination of original channels and new channel are fed into EM algorithm for video epitomic analysis. The edges can be preserved better than that without detail channel, just like the comparison in Figure 6(c) and Figure 6(d).

IV. THE PROPOSED INTRA CODING METHOD

After video epitomic analysis, image epitome for this video shot is obtained and is buffered in the memory. We are inspired by the new techniques such as intra displacement compensation (IDC) [4] and template matching prediction (TMP) [3] in intra coding for single layer. The IDC technique should code the motion vectors explicitly, and then can be applied to 8x8 block in new coding method. But the TMP technique does not need to code motion vector, and then can be applied to 4x4 block to generate prediction in new coding method. The reason for such design is that the size of 8x8 block is too big for TMP and the prediction accuracy is not guaranteed for samples in 8x8 block which are distant from template, and the overhead of coding 16 4x4 blocks is too big for IDC. Instead of the reconstructed part of current frame in conventional IDC and TMP, image epitome is taken as the reference frame, and two additional modes should also be covered by H.264 syntax. The two new modes are combined with existing prediction modes to form candidate mode set, and the best prediction mode is selected from the set by rate-distortion optimization (RDO). Taking H.264/AVC for example, the detailed design is as follows:

For 4x4 block, the strategy of re-using of DC mode is adopted because there are 16 4x4 blocks in every macroblock, once new mode is introduced beyond existing 9 modes, the increasing overhead would be un-affordable. The prediction generation method is distinguished by the variance $\delta_p$ of adjacent boundary 8 pixels which is determined by equation (11), and when $\delta_p < Th$, conventional DC mode is used to generate prediction, but when $\delta_p > Th$, for the new DC prediction mode, the similar way with conventional template matching prediction is used to generate the prediction for current block, as the large variance indicates less probability to select conventional DC mode to get prediction. The reconstructed L-block adjacent to current block is taken as template, and through template matching, the relative neighboring block of the searched L-block is considered as the prediction. The main difference of our approach with conventional template matching prediction is that image epitome replaces the reconstructed part of current frame for the search and prediction generation process, as is illustrated by the left part of Figure 9. Another difference involves that in our approach 4x4 block is used while the template matching prediction in [3] using 2x2 block. The template in our approach is adjusted according to the availability of adjacent blocks. $Th$ is determined by quantization step, and is shown by equation (12).

$$\sigma_p = \left[ \sum_{k=1}^{l} (\mu - u_k)^2 + \sum_{k=0}^{l} (\mu - l_k)^2 + 4 \right] / 8$$

$$Th = \left[ \frac{Qstep^2 + 8}{16} \right]$$

Figure 8. The determination of the usage of DC mode
For each video shot, image epitome should be encoded and transmitted to the decoder side. In our proposed intra coding method, lossless image coding is adopted to encode image epitome. In the experiments, Minimum-Rate Predictors (MRP) proposed by Ichiro Matsuda et al. [28] is used to compress image epitome due to its excellent performance, and the lossless compression ratio can reach more than 6 times. Additional bits are required to indicate the integration of image epitome with video bit-stream, but this part of overhead can be neglected for the whole bit-rate statistics.

Intra refresh is a simple and useful method for error resilience. Some frames or macroblocks are selected to be coded with intra-frame or intra-macroblocks, preventing temporal propagation of transmission errors. Intra frame refresh is not usually applied into practice because the sudden increase of bit-rate which the intra frame brings would have great negative impact on the multimedia system. Intra macroblock refresh are used more commonly, and different intra macroblock refresh have been proposed in current literature, and consists of random intra refresh (RIR), intra refresh based on error tracking, and adaptive intra refresh (AIR), constrained random intra refresh etc. But for these intra refresh methods, intra coded macroblock (intra refresh) may depend on adjacent macroblocks whose information are used to predict current block. Once error happened at adjacent macroblocks (e.g. adjacent P macroblock’s reference frame is lost during transmission), intra refresh performance would be degraded greatly. The proposed intra coding of 8x8 block breaks up the dependence on adjacent macroblocks, and can further improve the error resilience of intra refresh.

**V. EXPERIMENTAL RESULTS**

### 5.1. Intra coding efficiency

In H.264/AVC reference model JM12.4 [21], we have realized the algorithm described in the fourth section. Image epitome is derived using the method in the third section. To simply the performance evaluation of proposed intra coding method, the size of image epitome is fixed to 64 (the height) \times 80 for QCIF (96x120 for CIF, 72x96 for QWGA), and the dimension of image patch is set to 8x8. To get the image patch set for coherence constraint, image patch sampling period in the original image is set to 0.5, which means that the image patches are sampled at an interval of (4,0), (0,4), or (4, 4) on image epitome. The 8x8 blocks partitioned in compression compose patch set for incompleteness term. ANN library [27] is used to do complexity-expensive patch searching. For the EM algorithm, 10 times are iterated to get image epitome. The Figure 18 in appendix A shows image epitome obtained through our modified method and the method in [17] respectively for different test sequences. Image epitome is coded with the method of Minimum-Rate Predictors (MRP). The lossless coding results are shown in Table.2.

#### Table.1. Intra coding test conditions

<table>
<thead>
<tr>
<th>Configuration option</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>JM version</td>
<td>12.4</td>
</tr>
<tr>
<td>Profile</td>
<td>HIGH PROFILE</td>
</tr>
<tr>
<td>IntraPeriod</td>
<td>1 (all I frame)</td>
</tr>
<tr>
<td>FramesToBeEncoded</td>
<td>150</td>
</tr>
<tr>
<td>RDOPerformance</td>
<td>On</td>
</tr>
<tr>
<td>IntraMode</td>
<td>4×4, 8×8</td>
</tr>
<tr>
<td>Quatization parameter</td>
<td>32, 36, 40, 44, 48</td>
</tr>
<tr>
<td>Test sequence</td>
<td>QCIF: Foreman Silent Container Mobile Hall Tempete CIF: Foreman Hall QWGA: Keiba BOSSquare</td>
</tr>
</tbody>
</table>

The first experiment is taken under the configurations of Table.1. The sequences are coded with all-I frame mode. We calculate the average PSNR gain at the same bit-rate and the equivalent average bit-rate saving at the same PSNR using the method in [22], and the experimental results of average PSNR gain and equivalent bit-rate saving for all the sequences are described in Table.3, while the R-D curves for some representative sequences are shown in Figure 19. (in appendix B) in which ‘tmp’ means only new coding mode of 4x4 block is used, and ‘tmp+idc’ means both the new mode of 4x4 block and the new mode of 8x8 block are used. For the test sequence Foreman, the coding gain is the most obvious the average PSNR gain for all QP reaches about 1.86dB, and the lowest coding gain is achieved on sequence Mobile, which is only 0.11dB. The average PSNR gain for all test sequences reaches 0.64dB. In the meantime, the comparison experiments are done to evaluate our method (denoted by ‘epitome’ in Figure 11) and T.K Tan’s template matching prediction method [3] (denoted by ‘TMP’ to be distinguished with ‘tmp’ in Figure 18), and the results are illustrated by Figure 11. Our method outperform TMP by about 0.2dB.
Table 2. Lossless coding results of image epitome

<table>
<thead>
<tr>
<th>Test sequence</th>
<th>image epitome's size (bits)</th>
<th>Test sequence</th>
<th>image epitome's size (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman (QCIF)</td>
<td>16710</td>
<td>Tempete (QCIF)</td>
<td>22398</td>
</tr>
<tr>
<td>Silent (QCIF)</td>
<td>16982</td>
<td>Foreman (QCIF)</td>
<td>37382</td>
</tr>
<tr>
<td>Container (QCIF)</td>
<td>23422</td>
<td>Hall (QCIF)</td>
<td>47542</td>
</tr>
<tr>
<td>Mobile (QCIF)</td>
<td>23966</td>
<td>Keiba (QWGA)</td>
<td>22438</td>
</tr>
<tr>
<td>Hall (QCIF)</td>
<td>21062</td>
<td>BQSquare (QWGA)</td>
<td>30910</td>
</tr>
</tbody>
</table>

Table 3. Experimental results

<table>
<thead>
<tr>
<th>Sequence</th>
<th>△PSNR gain (dB)</th>
<th>△bit-rate saving (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q CIF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Silent</td>
<td>0.40</td>
<td>6.61</td>
</tr>
<tr>
<td>Hall</td>
<td>0.65</td>
<td>7.33</td>
</tr>
<tr>
<td>Foreman</td>
<td>1.86</td>
<td>24.15</td>
</tr>
<tr>
<td>Mobile</td>
<td>0.11</td>
<td>1.28</td>
</tr>
<tr>
<td>Container</td>
<td>0.89</td>
<td>12.27</td>
</tr>
<tr>
<td>Tempete</td>
<td>0.32</td>
<td>4.27</td>
</tr>
<tr>
<td>CIF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Foreman</td>
<td>1.12</td>
<td>17.71</td>
</tr>
<tr>
<td>Hall</td>
<td>0.57</td>
<td>7.59</td>
</tr>
<tr>
<td>QWGA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keiba</td>
<td>0.20</td>
<td>3.52</td>
</tr>
<tr>
<td>BQSquare</td>
<td>0.29</td>
<td>4.93</td>
</tr>
<tr>
<td>Average</td>
<td>0.64</td>
<td>8.97</td>
</tr>
</tbody>
</table>

To make the experimental comparison more close to practical applications, the second experiment is done on IPP coding pattern. The I frame appears at every 15 frames, and the other experimental conditions are set the same with the first experiment. The sequence Foreman and Tempete which achieve high and low coding gain for all-I frame mode respectively are selected for this evaluation. The coding gain on Foreman in term of PSNR still can reach about 0.2dB, but the one on Tempete is minor. The R-D curves are shown in Figure 10.

From all the above experimental results, we can observe some phenomenon for the proposed intra coding method which needs our more concerns:

1. The sequence Foreman has clear textures and sharp edges, and image epitome can preserve enough texture and shape information, which is very useful for improving the coding performance. For other sequences, the edge of the image is not obvious, that is, high-frequent component is not dominant in the image. In the process of generating image epitome, image epitome is made blurry due to continuous updating and mixing of different patches.

2. In the sequence Foreman, there is no camera zooming, so the mapping from the image epitome and original image is accurate. But for some other sequence, for example, the sequence Mobile, there is a continuous camera zooming out so that the prediction from image epitome is not accurate.
For intra coding with epitomic priors can apply temporal prediction in an implicit way for I frames, in real applications, the first I frame can also be taken as reference start point for intra refresh (a.k.a. P frame taking first I frame as reference). The advantage of image epitome lies in its coverage of various video content in a compact way, and only the first frame (or any other single frame) can not guarantee to include all the features in the video. In this experiment, four different QCIF sequences (Foreman, Container, Silent, and Hall) are combined to form a new QCIF sequence with four video scenes. The epitome of this new sequence is derived with size of 176x144 (illustrated in Figure.12), and is coded by MRP. The new proposed coding method is applied to the new sequence in all-I frame mode. The other coding method is P frame taking first I frame as reference, and only the first frame is coded with I frame while other frames are predicted taking the first frame as reference, and for this coding method, there are no extra bits for side information like epitomic priors. The results are shown in Figure.12. Our proposed method outperforms anchor method by up to 0.5dB. Though the first frame can provide very good prediction for only one of video scenes in video, it does not match other video scenes.

The third experiment is done to evaluate the coding performance of different video epitomic analysis methods. Different image epitomes on the first frame of Foreman are got by N.Jojic’s method, cropping method, and compression-oriented video analysis. The R-D curves are shown in Figure 13. In the figure, ‘c-epitome’ denotes using our proposed method to generate image epitome, and ‘epitome’ for N.Jojic’s method, and ‘cropping’ for cropping method. Compression-oriented video epitomic analysis can get the highest coding gain. Considering the epitome got by the ‘cropping’ method is the part of the first frame, this experiment can be also considered as the comparison of intra coding with compression-oriented video epitomic analysis with inter-I-frame prediction in which the first I frame is taken as reference frame.

5.2. The performance of error resilience

We evaluate the performance of error resilience for our new intra coding method (new intra coding of 8x8 block) using packet loss simulator of IP network [23], and the test sequence Foreman and Silent are coded with IPP pattern while I frame appears in every 60 frames. For RIR, the intra refresh strength is 10 intra macroblocks every frame. Figure 14. shows the curves of PSNR versus bit-rate in decoder side. In the figure, ‘RIR’ and ‘RIR-E’ mean the results with random intra refresh and the random intra refresh with epitomic priors respectively, and ‘C-RIR’ and ‘C-RIR-E’ mean the results with constrained random intra refresh and that with epitomic priors respectively. Due to the greatest importance of epitome, it is transmitted with the highest protection priority.
Figure 14. The R-D curves of different intra refresh methods under different PLR.

Figure 15. The PSNR fluctuation of every frame of Silent in decoder side under 5% PLR at the bit-rate of 260kbps with different intra refresh methods.
From the results shown in Figure 14 and 15, we can see no matter RIR or C-RIR is used, epitomic priors can improve the PSNR in the decoder by up to nearly 2dB with the proposed method. The subjective quality of decoded pictures can also be improved, which is shown in Figure 16.

5.3. The performance of spatially scalable video coding

To compare with our algorithm, the scaling, cropping and seam carving algorithms are also implemented. From the experimental results shown in Figure 17, we find that epitomic resizing algorithm not only can preserve the global information in the original image, and make a good trade-off between the completeness and coherence of resized image, but also can emphasize the image quality for regions of interest.

We also have applied epitomic resizing into spatially scalable video coding, and implemented the inter-layer prediction method depicted in section 4 on the platform of JSVM9.14 [25], and the new codec is named as E-SSVC hereafter. In the experiments, the scalable video bit-stream consists of two spatial layers: CIF for the enhancement layer and QCIF for the base layer. The input base layer video for conventional JSVM is derived through scaling (low-pass filtering and down-sampling), and that for E-SSVC is derived from epitomic resizing. To simplify the experiments, we only encode 5 frames using intra-coded way. CABAC is used in all codecs. The PSNR and bit-rate for enhancement layer are shown in Table 4. The PSNR and bit-rate for base layer of coastguard are 35.650dB and
### Table 4. Experimental results

<table>
<thead>
<tr>
<th>QP</th>
<th>Coastguard</th>
<th></th>
<th>Tempe</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>JSVM</td>
<td>E-SSVC</td>
<td>Single-layer</td>
<td>JSVM</td>
</tr>
<tr>
<td>30</td>
<td>PSNR (dB)</td>
<td>34.275</td>
<td>34.236</td>
<td>33.993</td>
</tr>
<tr>
<td></td>
<td>Rate (bpp)</td>
<td>1.182</td>
<td>1.158</td>
<td>0.873</td>
</tr>
<tr>
<td>32</td>
<td>PSNR (dB)</td>
<td>32.819</td>
<td>32.756</td>
<td>32.548</td>
</tr>
<tr>
<td></td>
<td>Rate (bpp)</td>
<td>0.977</td>
<td>0.962</td>
<td>0.686</td>
</tr>
<tr>
<td></td>
<td>Rate (bpp)</td>
<td>0.828</td>
<td>0.822</td>
<td>0.542</td>
</tr>
<tr>
<td>36</td>
<td>PSNR (dB)</td>
<td>30.469</td>
<td>30.387</td>
<td>30.080</td>
</tr>
<tr>
<td></td>
<td>Rate (bpp)</td>
<td>0.694</td>
<td>0.699</td>
<td>0.410</td>
</tr>
</tbody>
</table>

0.633bpp for E-SSVC, and those for JSVM are 34.021dB and 0.867bpp. The PSNR and bit-rate for base layer of Tempete are 34.469dB and 1.360bpp for E-SSVC, and those for JSVM are 33.703dB and 1.503bpp. From the above experimental results, we find that E-SSVC can outperform conventional JSVM by up to 0.2dB at the same bit-rate.

#### 5.4. Computational complexity

The main computational complexity for this new algorithm is distributed in video epitomic analysis, and image epitome is computed through a large amount of iterations for patch searching. In our experiments, it takes about 30 minutes to obtain a 64 (height)x80 epitome from 150 QCIF (176x144) frames (CPU: Core Duo 2.4G, RAM:2G). There are several techniques to further reduce computational complexity: (1) Primary Component Analysis (PCA). The dimension of image patch is 96, and if detail channel is added, the dimension will increase to 160. If the dimension is reduced to 20–30, the search process will be accelerated. (2) The images in the same video shot has great similarity, so to simplify computation process of image epitome, only the first image can be fed into the computation process, and image epitome is considered for image epitome for the video shot, so the computation of image epitome become practical in real applications. For the encoder, given the specific image epitome, the complexity is increased by about 2–3 times through observation of encoding time due to the search for prediction generation. For the decoder, no extra complexity will be brought by the new mode of 8x8 block, and the new mode of 4x4 block will bring complexity increase, therefore, additional increase in computational complexity for decoder side depends on the proportion of the new mode of 4x4 block.

### VI. CONCLUSION

In this paper, a new intra coding and refresh method based on video epitomic analysis is proposed, and can significantly improve the performance of intra frame, and the average PSNR can be raised by about 0.64dB. At the mean time this new intra coding method can further enhance the performance of intra refresh. When applied to scalable video coding, the whole coding efficiency of spatially scalable video coding can be improved. For the future works, we will focus on the better image epitome generation method which can preserve more details and integration of affine motion model into image epitome prediction to adapt to the camera zooming in image content.

### ACKNOWLEDGEMENTS

The authors are grateful to Dr. Feng Wu and Dr. Xiaoyan Sun for their valuable suggestions. This research is supported by the National Natural Science Foundation of China under Grant No.60970160, No.61070080 and No.61003184; the National Grand Fundamental Research 973 Program of China under Grant No.2009CB320906; the Ph.D Candidates Self-research (including 1+4) Program of Wuhan University in 2008.

### REFERENCES

Appendix. A
Figure 18. The first frame of every sequence ((a) column), the sequence’s image epitome by N.Jojic’s method ((b) column), image epitome by compression-oriented video epitomic analysis ((c) column)
Figure 19. The detailed R-D curves of every test sequence in all I frames