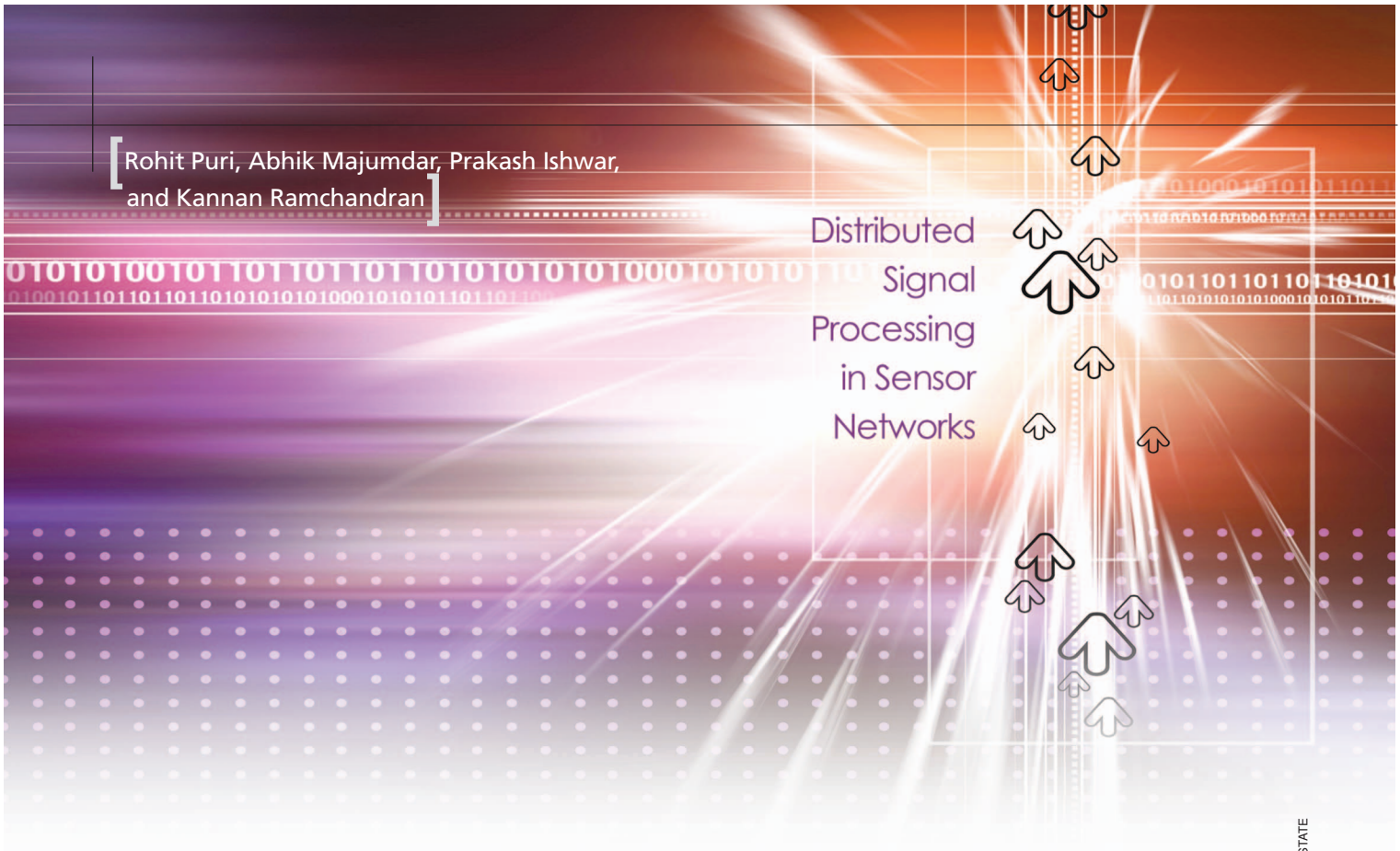


[Rohit Puri, Abhik Majumdar, Prakash Ishwar, and Kannan Ramchandran]

Distributed Video Coding in Wireless Sensor Networks

[Exploiting the spatiotemporal redundancy in broadband networks]

The vision of miniaturized computers, decked with tiny batteries, sensors, and radios, organizing themselves tetherlessly to efficiently and reliably perform complex tasks, considered unimaginable a few years ago, is no longer just a vision. It is rapidly evolving into the reality of sensor networks. Today, we are asking these networks to help us in innumerable ways. These range from detecting chemical leaks to tracking children in Disneyland, from monitoring factory inventories and managing company assets to even probing the ecosystem at unprecedented spatial scales. The rich application space addressed by these so-called first-generation sensor networks has understandably gotten us excited. The ability to throw a bunch of lightweight sensing devices into an arena of interest, and to reliably and efficiently monitor (and sometimes even control) sensing modalities like light intensity, temperature, pressure, chemical concentration, and magnetization, is indeed of great value.

Most of these sensing modalities correspond to low data-rate signals. Indeed, this is tough enough given the daunting challenges posed by energy management, wireless interference,

1053-5888/06/$20.00©2006IEEE

and ad hoc networking. However, there is a growing base of compelling applications that demand supporting communication of high data-rate visual information over wireless networks. In scenarios where we have little or no prior knowledge of what information in an environment may prove critical, it is essential to have access to "raw" visual data for more informed real-time decision making and postevent data analysis. For example, while pressure and magnetic sensors can register the presence or absence of a car in a parking lot, they cannot help identify the drivers, accident victims, witnesses, and vehicle license plates in unexpected accident situations. In such situations, the data gathered by a network of video sensors becomes indispensable.

The deployment of high-speed, wired and wireless networks such as 802.16, 802.16a, and 802.11b/g and the explosion of digital camera equipped cellular phones has already provided basic infrastructure for supporting communications in high data-rate wireless video sensor networks [1]. These networks can find their way into many real-time applications needing video-based active monitoring of telemetry data in such diverse indoor and outdoor environments as hospitals, hotels, parking lots, highways, airports, and international borders. Typical video sensor networks are made up of multiple cameras with varying degrees of spatially and temporally overlapping coverage, generating correlated signals that need to be processed, compressed, and exchanged in a loss-prone wireless environment to facilitate real-time decisions. However, the sheer volume of visual data involved, with video signals ranging from a few hundreds of kilobits per second to a few megabits per second and more, poses new and unique challenges.

This forms the motivational background for this article. Can we make progress towards a second generation of broadband sensor networks that cover applications like video? There are numerous challenges to be addressed to take this to reality. At the very least, we need to address the first-generation challenges that get amplified by the increased data rate and energy requirements. Further, interdisciplinary research involving tools from diverse areas such as computer vision, video processing, distributed computing, and broadband wireless networks is in order.

In this article, we address the important aspect of compressing and transmitting the video signals generated by these broadband networks while heeding the architectural demands imposed by these networks in terms of energy constraints (communication and computation) as well as the channel uncertainty related to the wireless communication medium. (For issues related to the interaction of signal processing and networking, see, for example, [2].) To take an even smaller and manageable bite out of this daunting challenge, we concentrate here on the exemplary case of a single video camera and use it as a platform to describe the theoretical principles and practical aspects underlying distributed video coding. The extension of these concepts to the general multicamera video sensor network environment remains an ongoing challenge for the video networking community at large at this time, with promising preliminary efforts by several R&D groups [3]–[6]. The first step involves a thorough understanding of the fundamental issues underlying a simple single-camera point-to-point setup. We argue that the key concepts of distributed video coding for networks can be extracted even from this setup and are largely independent of the number of nodes in the system. The primary intent of this article is to expose these concepts in a fundamental way. The networked multicamera case will be addressed more tersely later in the article, with the tacit acknowledgment that this is very much a fledgling area of research that will need to build a critical mass in the coming years to bring to fruition.

A broadband network of wireless video sensors is subjected to three principal constraints:

1) *limited processing capabilities* and diverse display resolutions due in part to inexpensive device designs and limited battery power. These call for lightweight signal processing and compression algorithms at the individual sensor nodes and an architecture that can adapt to the differing processing capabilities of the encoding and decoding nodes.

2) *limited power/energy budget* requiring careful management for maximizing network lifetime, the quality of the acquired data, and the accuracy of the decisions. Communication is often the dominant power-consuming activity. Power management requires efficient compression algorithms that maximize the power utilization per bit communicated and controlled dormancy cycles in intersensor communication that preclude frequent intersensor communication. This motivates the need for distributed coding and processing.

3) *information loss* that is endemic to the harsh, loss-prone, wireless communication environment. This calls for robust coding algorithms, communication and networking protocols, and architectures that are immune to single points of failure. It is important to proactively build in robustness considerations into the architectural foundation rather than as after-thought bandage fixes.

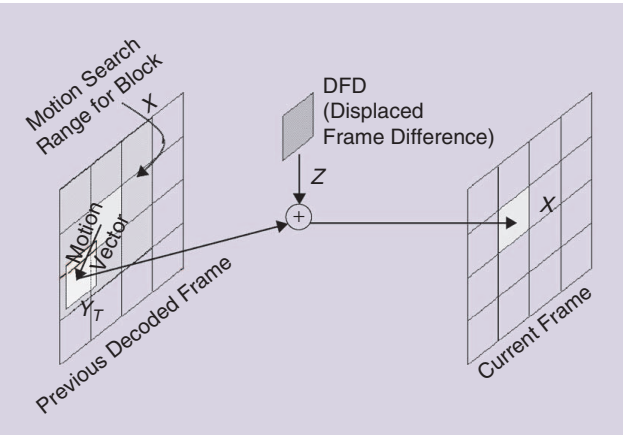
With the above constraints, the traditional views of video coding and transmission as being confined to a "downlink" scenario (such as television broadcast or download from a video server) need to be relaxed. In the prevalent video coding architectures such as MPEG-x and H.26x [7]–[10], video *encoding* is the primary computationally intensive task with the complexity dominated by the motion-search operation. Conventional video *decoding,* on the other hand, has significantly lower complexity. This skewed, somewhat rigid, complexity compartmentalization conflicts with the heterogeneous processing capability requirements of video sensor networks where the encoding units might be able to do only "lightweight" processing but the relay or decoding units might be more capable. The prevalent video coding architectures are also built upon the principle of (deterministic) predictive coding from which they derive their compression efficiency. However, as clarified later in the article, predictive coding architectures are prone to encoder-decoder drift due to dependency chains and are therefore fragile to transmission losses. Drift recovery requires bandwidth-expensive (intraframe mode) resets. The use of forward error correction

codes (FECs) can delay but not stem the onset of drift. Further, generalizing these deterministic predictive coding architectures to the multicamera setting is not easy without fairly elaborate intercamera communication, which can be expensive and complicated. Clearly, this does not scale well to large-scale distributed camera networks. In summary, the traditional video coding architectures are inherently mismatched to the challenging requirements imposed by the emerging class of video sensor network applications.

This article conducts a study of broadband video-based sensor networks including: i) the fundamental requirements imposed by these networks, ii) the theoretical foundations and architectural paradigm shifts needed to address these requirements effectively, iii) real-world experimental validation of the proposed architecture and algorithms, and iv) the wide spectrum of possible applications.

There is a high degree of spatiotemporal correlation in the data gathered by a broadband video camera network, and distributed source coding principles [11], [12] provide useful tools for efficiently exploiting this correlation. We will explore these aspects at theoretical, system-design, and algorithmic levels in the context of wireless video sensor networks. As motivated earlier, we will focus primarily on a single-camera setup to illustrate our ideas, highlighting the key differentiating attributes of distributed video coding that cannot be supported by existing video coding methods. As will become clear in what follows, the single camera setup constitutes a key building block of the video sensor network with direct applications to scenarios such as wireless networks of video camera equipped cell phones. Further, the proposed methods scale naturally to the multiple-camera scenario.

We would like to point to the recent heightened interest and spate of research activity in the area of video coding with side information (distributed video coding) as in [13]–[14]. We will confine ourselves here to the PRISM codec of [13], which leverages the power of distributed compression methods [12] to achieve superior robustness to frame drops at very low delays with low encoding complexity (of the order of still image compression) and competitive compression performance (see [16] for details).



[FIG1] P-Frame coding (motion-compensated predictive video coding): The current frame is divided up into blocks of *n* pixels. X is the current block being encoded. $Y_1, \ldots, Y_M$ are *M* candidate predictor blocks for X in the previous decoded frame within a search range. Y_T is the best predictor for X. Z corresponds to the prediction error (or innovations noise).

This article is organized as follows. The next section overviews the conventional interframe predictive video coding architecture. That is followed by a description of the architectural goals and the basic philosophy underlying the proposed PRISM framework. Then information-theoretic performance limits of prediction-based and side-information based video codecs under tractable semirealistic models for the source and channel impairments are presented. Analysis of pure compression performance reveals the novel feasibility of moving the high-complexity predictive motion search task from the encoder to the decoder. Analysis of channel impairments reveals the fundamental unsuitability of predictive coding in loss-prone environments. These theoretical insights guide the experimental results presented. Finally, we conclude this article with a look into the multicamera setup, and some exciting directions for the future.

MPEG: A PROTOTYPICAL ARCHITECTURE FOR PREDICTIVE VIDEO CODING

This section quickly overviews the conventional video interframe predictive coding architecture that underlies current video coding standards such as the MPEG-x and H.26x. Video is a temporal sequence of two-dimensional images (also called frames). For the purpose of encoding, each of these frames is partitioned into regular spatial blocks. These blocks are encoded primarily in the following two modes.

1) *Intracoding (I) mode:* The intracoding mode exploits the spatial correlation in the frame that contains the current block by using a block transform such as the discrete cosine transform (DCT). It typically achieves poor compression, since it does not exploit the temporal redundancies in video.

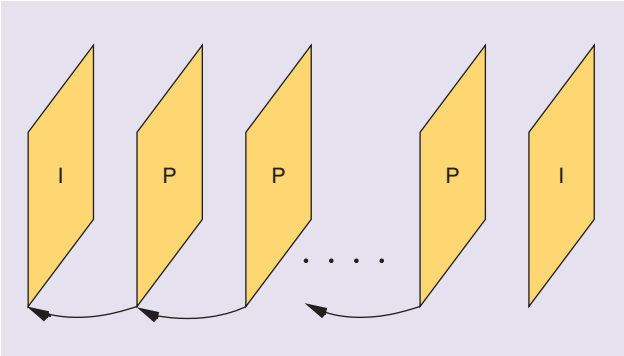
2) *Intercoding or motion compensated predictive (P) mode:* This mode exploits both the spatial and temporal correlation present in the video sequence resulting in *high compression*. The *high-complexity* motion estimation operation uses the frame memory to infer the best predictor block for the block being encoded. Motion compensation provides the residue between the predictor block and the block in question, which is then transformed and encoded. Intercoding is illustrated in Figure 1.

Typically, the video sequence is grouped into a group of frames (GOF) (see Figure 2) where the first frame in the group is coded in intramode only while the remaining frames in the group are usually coded in intermode.

Intracoding has low encoding complexity and high robustness (being a self-contained description of the block being encoded) but has poor compression efficiency. To offset this, the MPEG-x and H.26x standards use motion compensated predictive coding to achieve the compression needed to communicate over bandwidth-constrained networks. However, motion compensated predictive coding suffers from two major drawbacks:

a) Fragility to synchronization or "drift" between encoder and decoder in the face of prediction mismatch, primarily, due to channel loss, is a major drawback of the current paradigms. (Difference in frame memories at the encoder and the decoder results in the residue error being encoded at the encoder off some predictor and decoded at the decoder off some other predictor causing drift. This is a major problem in wireless communication environments that are characterized by noise and deep fades. Scenarios such as transmission losses can lead to nonidentical encoder and decoder frame memories.)

b) These frameworks are hampered by a rigid computational complexity partition between encoder (heavy) and decoder (light) where the encoding complexity is dominated by the motion search operation, whereas the decoder is a lightweight device operating in a "slave" mode to the encoder. (A "full-search" block motion estimation algorithm incurs approximately 65,000 operations per pixel per second for a 30 frames per second video and can consume nearly 75% of CPU time. Motion estimation is also extremely demanding on the I/O transfer between CPU and memory.)



[FIG2] A GOF. Here I = intracoded frames and P = motion-compensated intercoded frames.

PRISM: A NEW ARCHITECTURE FOR DISTRIBUTED VIDEO CODING

As discussed earlier, wireless video sensor networks are characterized by devices with limited processing capabilities and battery power constraints, harsh loss-prone wireless channels, and (comparatively) low bandwidths. Consequently, a video codec designed for a wireless video sensor network is desired to have

- inbuilt robustness to "drift" caused by loss of synchronization between encoder and decoder (e.g., due to channel loss)
- flexibility in the distribution of computational complexity between encoder and decoder
- high compression efficiency.

In addition, some applications impose very stringent delay requirements. Conventional video codecs, such as MPEG-x and H.26x, fail to meet all these requirements simultaneously. In the sequel, we will describe the architectural and algorithmic aspects of PRISM, which is grounded on the framework of source coding with side information (also called distributed source coding). We illustrate this concept in "Illustrative Example for Coding with Side Information"

ILLUSTRATIVE EXAMPLE FOR CODING WITH SIDE INFORMATION

To see how source coding with side information (Wyner-Ziv coding) works in practice, it is instructive to examine the following example. Here X is a real-valued observation at the encoder that has to be communicated to the decoder with a certain fidelity. The decoder has access to correlated side-information Y which is not available at the encoder.

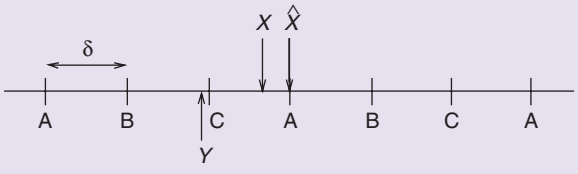
The encoder will first quantize X to $\hat{X}$ with a scalar quantizer with step size δ (Figure 3). Clearly, the distance between X and $\hat{X}$ is bounded as $|X - \hat{X}| \leq \delta/2$. We can think of the quantizer as consisting of three interleaved quantizers (cosets), each of step size 3δ. In Figure 3 we have labeled the reconstruction levels of the three quantizers as A, B, and C, respectively. The encoder, after quantizing X, will note the label of $\hat{X}$ and send it to the decoder, which requires $\log_2(3)$ bits.

The decoder has access to the label transmitted by the encoder and the side information Y. In this example, we assume that X and Y are correlated such that $|Y - X| < \delta$. Thus, we can bound the distance between $\hat{X}$ and Y as

$$|\hat{X} - Y| \leq |\hat{X} - X| + |X - Y| < \frac{\delta}{2} + \delta = \frac{3\delta}{2}.$$

Because $\hat{X}$ and Y are within a distance of $(3\delta)/2$ of each other and the reconstruction levels with the same label are separated by 3δ, the decoder can correctly find $\hat{X}$ by selecting the reconstruction level with the label sent by the encoder that is closest to Y. This can be seen in Figure 3, which shows one realization of X and Y.

In this example, the encoder has transmitted only $\log_2(3)$ bits per sample, and the decoder can correctly reconstruct $\hat{X}$, an estimate within $\delta/2$ of the source X. In the absence of Y at the decoder, the encoder would need to quantize X on an m-level quantizer of step size δ. Thus, by exploiting the presence of Y at the decoder, the encoder saves $(log_2(m) - log_2(3))_+$ bits—this can be quite large if m is large, which should be the case if the variance of X is large.



[FIG3] Distributed compression example: The encoder quantizes X to $\hat{X}$ and transmits the coset-label "A" of $\hat{X}$. The decoder finds the coset-A reconstruction level closest to the side information Y as $\hat{X}$.

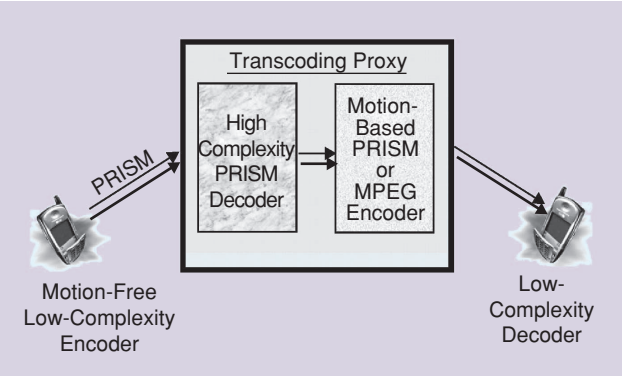
and will discuss the three major architectural goals of PRISM in detail in the context of the example.

COMPRESSION PERFORMANCE

An intuitive explanation of the example in "Illustrative Example for Coding with Side Information" is that the source quantizer is partitioned into cosets of a channel code [17], [18] (the three interleaved quantizers A, B, and C). The side-information Y can be viewed as a free (but noisy) version of the source X available at the decoder. The decoder decodes this noisy version of X in a channel codebook (the specific codebook used will be the coset specified by the encoder). In the example, we have used a channel code that is "matched" to the correlation distance (equivalently, noise) between $\hat{X}$ and Y to partition the source code-word space of X. This provides a side information codec with high compression performance. In theory, for many interesting scenarios, the performance of side-information coding system can match that of one based on predictive coding (where both the encoder and the decoder have access to Y) [11], [12].

FLEXIBLE DISTRIBUTION OF COMPLEXITY

A second goal is to allow for the complexity burden between encoder and decoder to be shared in any desirable ratio as demanded by prevailing channel conditions and the constraints of the encoding and decoding devices, without loss of performance. Since the predominant complexity component in current state-of-the-art video encoders is the motion estimation function, PRISM facilitates the above goal by allowing for moving the expensive motion search component to the decoder. (Here we focus primarily on the two extremes: the entire motion search operation being performed either at the encoder or the decoder. However, the PRISM framework accommodates arbitrary sharing of the motion search space between the encoder and the decoder. For instance, the encoder can perform a coarse-level motion search and reveal the outcome to the decoder. The decoder can then limit its search to the complement of the encoder search set thereby reducing its search cost.) This is based on a generalization of the source coding with side information framework where there is uncertainty in the state of side information at the receiver [19].



[FIG4] System-level diagram for a network scenario with low-complexity encoding and decoding devices.

We note that, while in theory, it can be possible to move motion search complexity between encoder and decoder with no loss of performance; in practice, the correlation structure between the current data and the predictor information isn't known a priori and the encoder needs to expend work to find this out. In terms of the example in "Illustrative Example for Coding with Side Information," the encoder would need to invest some resources to find out that $\hat{X}$ and Y differ by at most $3\delta/2$. Hence, the well-known complexity-compression tradeoff (the richer the motion model, the better the compression) witnessed in conventional video codecs is also observed in the PRISM framework with a low-complexity PRISM encoder taking a hit in compression performance relative to an interframe codec (about $1-1.5$ dB as in our simulation results).

Interestingly, the amount of motion search needed at encoder and decoder also depends on channel conditions. Specifically, as the channel noise increases, doing a full-motion search at the encoder gives diminishing marginal utility over doing a coarse-grained motion search. On the other hand, as the channel degrades, the decoder will need to search more among the list of available predictors to find one that enables successful decoding.

ROBUSTNESS

A major goal of PRISM is to allow for far greater robustness to packet and frame drops than is possible with today's video codecs. PRISM targets this by using the "universally robust" side-information based coding framework. The partitioning of X in "Illustrative Example for Coding with Side Information" is *universal* in the sense that the same partitioning of X works for all Y regardless of the value of Y as long as both X and Y satisfy the correlation structure s.t. $|\hat{X}-Y|<3\delta/2$.

Essentially, in the predictive coding framework the encoding for the current unit hinges on a single deterministic predictor, the loss of which results in erroneous decoding and error propagation. On the other hand, a side-information coding based paradigm encodes the current unit, in principle, with respect to the *correlation statistics* between the current unit and the predictor only. At the decoder, the availability of *any* predictor that satisfies the correlation statistics enables correct decoding.

PRISM IN A NETWORK CONFIGURATION

In a video sensor network, it is possible that both the encoding and decoding devices have limited processing capabilities. In this case, using the flexible complexity partitioning feature of PRISM, it is possible to move the computational burden to an intermediate network node by using a transcoding proxy as shown in Figure 4.

TOWARDS AN INFORMATION THEORY FOR DISTRIBUTED VIDEO CODING

Earlier, a distributed compression paradigm was proposed as a promising approach for meeting the challenging requirements imposed by wireless video sensor networks. This section presents theoretical constructions and related analytic studies that serve to

clarify, quantify, and demonstrate the theoretical feasibility of meeting the proposed objectives. These theoretical results employ tools from multiuser information theory [20] together with models that aim to strike a balance between analytical tractability and the realism of real-world video signals. In this context, it should be noted that there is a disconnect between video coding practice and the information-theory prescriptions described in this section in terms of a) the stringent delay and complexity requirements of the former and the asymptotically large coding delays permitted by the latter and b) the use of relatively simple models that do not capture the rich and complex video phenomena in their entirety. Even so, the proposed models and information-theoretic analysis capture the essence of the problem at hand and offer valuable insights that can be directly translated to implementable practical algorithms. An information-theoretic analysis also provides quantitative performance bounds.

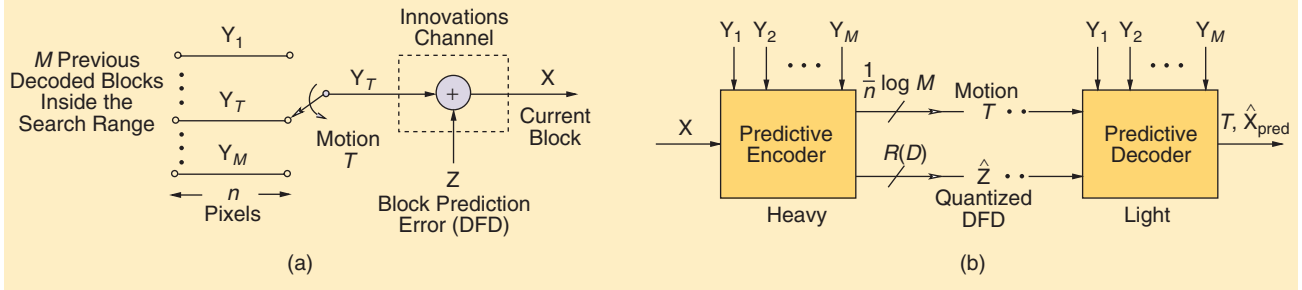
MOTION-COMPENSATED PREDICTIVE VIDEO CODING

The existing video coding architectures like MPEG-x and H.26x have been optimized to operate in practically *error-free communication* environments with maximum compression efficiency and low decoding complexity. The high compression efficiency is achieved primarily by exploiting the strong temporal correlation between successive video frames through motion-compensated predictive coding (MCPC) illustrated in Figures 1 and 5. A block of n pixels $\mathbf{X}$ in the current frame is approximated using the best mean squared error (MSE) matching predictor block $\mathbf{Y}_T$ (out of M candidate predictors $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$) inside a search region that is spatially close to the location of $\mathbf{X}$ but in the previous decoded frame. The *parameter* $T \in \{1, \ldots, M\}$, called motion index, represents an ambiguous state of nature that accounts for motion between consecutive frames. The estimated T is first sent to the decoder using $\log_2(M)/n$ bits per pixel (b/p). If the encoder and decoder are synchronized (no previous transmission losses), the decoder will have the same previous decoded blocks $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ as the encoder. Once the decoder knows T, the video coding problem is reduced to the problem of compressing the "source" $\mathbf{X}$ using the correlated side-information $\mathbf{Y}_T$ now available to *both* the encoder and the decoder. The optimum solution to this problem is well known: In the second step, the prediction error (or innovations noise) $\mathbf{Z}$, which is roughly independent of $\mathbf{Y}_T$, should be quantized to the nearest code word $\hat{\mathbf{Z}}$ in a shared rate-$R(D)$ optimum rate-distortion codebook for $\mathbf{Z}$ and sent to the decoder. The decoder upon receiving the quantized code word should reconstruct the source block as $\hat{\mathbf{X}}_{\text{pred}} = \mathbf{Y}_T + \hat{\mathbf{Z}}$ and thereby achieve a distortion D. The total rate needed is $(\log(M)/n) + R(D)$ b/p. This gives the optimum rate versus distortion performance [19], [21]. Specifically, if the components of $\mathbf{Z}$ have independent and identically distributed Gaussian statistics with variance σ_z^2, $R(D)$ is given by [20]

$$R(D) = \min\left(0,\, 0.5 \log\left(\sigma_z^2/D\right)\right). \tag{1}$$

(With video coders moving towards increasingly sophisticated motion models, if motion compensation was perfect, $\mathbf{Z}$ will truly appear as white noise. Gaussian statistics for $\mathbf{Z}$ and a block-motion model is often assumed to simplify analysis and gain insight and also because the performance under Gaussian statistics often bounds the performance under other statistics having the same mean and variance. Sophisticated motion models, example, affine motion and optical flow, can also be handled within the scope of ideas described here.) For independent, white, Gaussian $\mathbf{Z}$, maximum likelihood (ML) estimation of T from $\mathbf{X}$ and $\{\mathbf{Y}\}_{i=1}^{M}$ coincides with the standard block-matching procedure of finding the MSE-optimal match for $\mathbf{X}$ among the $\mathbf{Y}_i$s. The ML estimate will be correct with high probability for large block-size n and "well-behaved" joint statistics of $\mathbf{X}$ and the $\mathbf{Y}_i$s.

Despite its excellent compression performance, MCPC suffers from the fragility to *mis-synchronization* or "drift" between encoder and decoder and a rigid skewed computational complexity partition between encoder (heavy) and decoder (light) that make it unsuitable for network-scaling in distributed environments. When the encoder is incapable of doing motion-estimation and compensation but the decoder/intermediate relay nodes having access to correlated side information $\mathbf{Y}_i$ s can "pick up the slack" (flexible distribution of computational complexity), how significant is the loss of rate-distortion coding performance over contemporary MCPC-based video codecs where encoders have the resources to make use of the correlated $\mathbf{Y}_i$s to encode $\mathbf{X}$? A surprising result is that the performance in both



[FIG5] (a) Motion-indexed additive-innovations model for video. (b) Motion-compensated predictive coding with a heavy encoder and a light decoder in the synchronized scenario. T is estimated and sent using $\log(M)/n$ b/p. The prediction error $\mathbf{Z}$ is quantized to $\hat{\mathbf{Z}}$ using $R(D)$ b/p and sent. $\hat{\mathbf{X}}_{\text{pred}} = \mathbf{Y}_T + \hat{\mathbf{Z}}$. The mean squared error is $D = \mathbb{E}||\mathbf{X} - \hat{\mathbf{X}}_{\text{pred}}||^2/n$.

scenarios is identical when the innovations process has Gaussian statistics and is independent of the correlated side information $\mathbf{Y}_i$s.

SHARING MOTION-COMPLEXITY BETWEEN ENCODER AND DECODER

NEW THEORETICAL FRAMEWORK FOR DISTRIBUTED VIDEO CODING

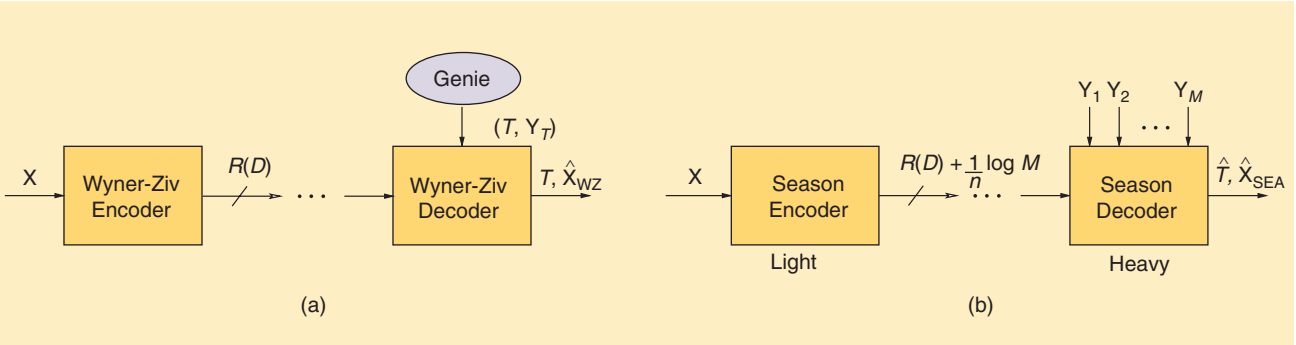
If an encoding node is incapable of performing the complex motion estimation, this is in effect pretending that the encoder does *not* have access to the previous decoded blocks $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ only the decoder has. The blocks $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ available only at the decoder represent *side-information* that is *statistically* correlated to the source $\mathbf{X}$ at the encoder. (In the unsynchronized case, the decoder's side-information would be corrupted.) This situation is temptingly similar to the famous problem of rate-distortion source coding with correlated side-information present only at the decoder [Figure 6(a)] which was completely solved by Wyner and Ziv [12]; but there is a catch. The joint statistics of the source and side-information is dependent on a state of nature T that is unknown to both the encoder and the decoder, that is, the underlying correlation is itself uncertain [Figure 6(b)]. This introduces a new dimension to the distributed source coding problem. Distributed compression has to be done *jointly* with inference unlike the situation in the classical problems of distributed source coding [11], [12] where the statistical models are assumed to be known perfectly. We dub this new coding framework as SEASON (Source Encoding with side-information under Ambiguous State of Nature) [19].

HELP FROM A GENIE OR WYNER-ZIV ENCODING-DECODING

Classical Wyner-Ziv coding exploits knowledge of the statistical correlation between the source and side-information to attain optimum rate-distortion performance. In the video context, the classical Wyner-Ziv coding situation corresponds to the encoder not having access to the $\mathbf{Y}_i$s and the decoder (and not the encoder) getting help from a Genie [19] who reveals the hidden state of nature T (and hence the joint statistics) as shown in Figure 6(a). For this Genie-assisted Wyner-Ziv coding situation of Figure 6(a) it turns out that if the prediction error $\mathbf{Z}$ is independent of the $\mathbf{Y}_i$s, is white and has Gaussian statistics, then the information-theoretic rate-MSE performance in the coding with side information situation is identical to the case when $\mathbf{Y}_T$ is also available to the encoder as in the previous subsection, that is, the minimum bitrate needed to achieve an MSE D is given by (1). Thus the Genie-aided Wyner-Ziv coding can essentially match the MSE performance of predictive coding using $R(D)$ b/p. Encoding proceeds by first designing a rate-distortion codebook of rate R' (containing $2^{nR'}$ code words) constituting the space of quantized code words for X. Each n-length block of source samples $\mathbf{X}$ is first quantized to the "nearest" code word in the codebook. As in "Illustrative Example for Coding with Side Information," the quantized code-word space (of size $2^{nR'}$ code words) is further partitioned into 2^{nR} cosets or bins ($R < R'$) so that each bin contains $2^{n(R'-R)}$ code words. This can be achieved by the information theoretic operation of random binning. The encoder only transmits the index of the bin in which the quantized code word lies and thereby only needs R bits/sample. The decoder receives the bin index and disambiguates the correct code word in this bin by exploiting the correlation between the code word and the matching n-length block of side-information samples $\mathbf{Y}_T$. Recovering the code word, it forms the minimum MSE estimate of $\mathbf{X}$ to achieve an MSE of D.

SEASON ENCODING-DECODING

However, the "real" situation is shown in Figure 6(b). In the Wyner-Ziv setup, upon receiving a bin-index, the decoder used the side-information $\mathbf{Y}_T$ to remove the uncertainty about the identity of the correct code word in the bin. However, in the absence of the Genie, T is not available to the decoder and represents an additional source of uncertainty [see Figure 6(b)]. However, it turns out that this additional uncertainty can be overcome by decreasing the size of the bins, that is, by having more bins. This incurs an additional bit-budget of $\log(M)/n$ b/p, the precise bit budget needed to convey the motion-index T to the decoder in the first step of predictive video coding. Encoding

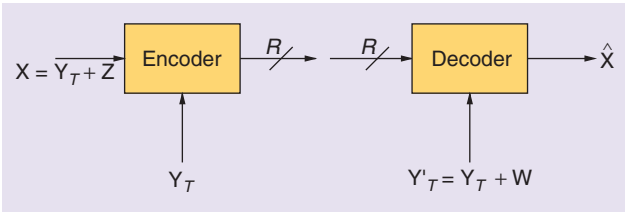


[FIG6] (a) Genie-assisted classical Wyner-Ziv coding. The Genie reveals $\mathbf{Y}_T$ only to the decoder. The encoder does not have access to or is constrained from using $Y_1, \ldots, Y_M$. (b) New distributed source coding framework under ambiguous state of nature with light encoder and heavy decoder. The rate-MSE performance matches optimal predictive coding and $\hat{T} = T$ with probability one for n large.

is relatively light and uses a Wyner-Ziv rate-distortion codebook as in the earlier Genie-assisted codec. But whereas earlier each bin contained $2^{n(R'-R)}$ code words, now each bin will contain $2^{n(R'-R-\log(M)/n)}$ code words. Upon receiving a bin index, the decoder is now faced with the task of discovering the code word sent by the encoder, without knowing T. The decoder tries each $\mathbf{Y}_i$ in turn and stops as soon as it has found a code word in the bin with which it is "sufficiently strongly correlated" according to the joint component statistics expected of $\mathbf{Y}_T$ and the quantized representation of $\mathbf{X}$. This heavy decoding procedure is like a "block-matching" motion-estimation operation but done at the decoding node [19], an unexplored notion in the traditional video community. (This is only to be consistent with the block-matching motion model for video that we have assumed for ease of illustration. Richer motion models involving complicated weighted combinations of intensities from multiple blocks from the previous frames can also be handled in a similar manner under the SEASON framework.) It can be demonstrated that this algorithm not only finds the correct quantized code word of $\mathbf{X}$, thereby matching the optimum rate-MSE performance of predictive codecs (for Gaussian innovations), but also recovers the correct motion-index T with high probability (for large block size n) [19]. This simple information-theoretic example reveals the potential for shifting the motion-complexity from the encoder to the decoder (without loss of coding performance) or absorbing it at capable intermediate relay nodes performing SEASON-decoding and predictive reencoding to support less capable origin and destination nodes.

ROBUSTNESS TO TRANSMISSION ERRORS

We shall now outline an information-theoretical analysis for a very simple mismatched side-information problem that clarifies the nature of the drift problem associated with predictive coding and will also highlight the superior robustness properties of distributed video coding. Consider the simplified Genie-assisted situation shown in Figure 7. Here, $\mathbf{X} = \mathbf{Y}_T + \mathbf{Z}$ is the data source that needs to be transmitted (the Genie has revealed T to the decoder), $\mathbf{Y}_T =$ predictor for $\mathbf{X}$ available at the encoder with associated (almost) independent innovations $\mathbf{Z}$, and $\mathbf{Y}'_T = \mathbf{Y}_T + \mathbf{W}$ is the predictor for $\mathbf{X}$ available at the destination. $\mathbf{W} \neq 0$ represents the *accumulated drift noise* unobservable at the encoder. The encoder and decoder are synchronized, and there is no drift if $\mathbf{W} = 0$. If $\mathbf{Y}$, $\mathbf{Z}$, and $\mathbf{W}$ are mutually independent and $\mathbf{Z}$ is Gaussian, it turns out that [22], [23] the optimum rate-distortion coding scheme is to *completely ignore* $\mathbf{Y}_T$ at the encoder, even though it is correlated to $\mathbf{X}$ and $\mathbf{Y}'_T$ and code in the Wyner-Ziv mode with $\mathbf{X}$ as source and $\mathbf{Y}'_T$ as correlated decoder side information. A predictive approach can first (phase 1) code the innovations $\mathbf{Z}$ to an MSE D as usual pretending that there is no side-information mismatch, that is, the decoder has $\mathbf{Y}_T$ and later (phase 2) spend additional bit rate to compensate for the resulting error due to nonzero drift ($\mathbf{Y}'_T \neq \mathbf{Y}_T$) or first (phase 2) spend rate trying to resynchronize the encoder and decoder side-information and then (phase-1) code $\mathbf{Z}$. This idea forms the basis of several practical distributed source-coding algorithms



[FIG7] Robustness analysis problem setup. A Genie has revealed T to the decoder. $W \neq 0$ models accumulated drift noise at the decoder.

for correcting drift in the recent distributed video coding literature [15], [24], [25]. Since the encoder does not know $\mathbf{W}$, it can be shown that the only way to achieve the information-theoretically optimal performance in phase 2 of the coding (the post-coding error compensation or precoding synchronization) is to use a distributed source coding approach to convey the missing information. Even when the purely predictive approach utilizes the optimal (necessarily) distributed approach in phase 2 to correctly decode $\mathbf{X}$, it incurs significant performance loss relative to the strategy that uses a single phase of purely distributed source coding which exploits the *statistical* correlation between $\mathbf{X}$ and $\mathbf{Y}'_T$. Specifically, if $\mathbf{Y}_T$, $\mathbf{Z}$, and $\mathbf{W}$ are independent, white, and jointly Gaussian, and $R_{\text{pred}}(D)$ and $R_{\text{distb}}(D)$ respectively denote the rates for the predictive (with phase 2 distributed coding) and single-phase purely distributed approach for meeting a target MSE D ($D < \sigma_z^2$) then (see [22] and [23])

$$R_{\text{distb}}(D) = \frac{1}{2}\log\frac{\sigma_z^2 + \frac{\sigma_y^2\sigma_w^2}{\sigma_y^2+\sigma_w^2}}{D} \tag{2}$$

where σ_y^2, σ_z^2, and σ_w^2 are the component variances of $\mathbf{Y}_T$, $\mathbf{Z}$, and $\mathbf{W}$, respectively, and

$$R_{\text{pred}}(D) = \frac{1}{2}\log\frac{\sigma_z^2\left(D + \frac{\sigma_y^2\sigma_w^2}{\sigma_y^2+\sigma_w^2}\right)}{D^2}.$$

The rate penalty for target MSE D is given by

$$R_{\text{pred}} - R_{\text{distb}} = \frac{1}{2}\log\frac{1 + A/D}{1 + A/\sigma_z^2},$$

where $A = \sigma_y^2\sigma_w^2/(\sigma_y^2 + \sigma_w^2)$. Since $D < \sigma_z^2$, the difference is strictly positive. (Note that $D < \sigma_z^2 =$ variance of $\mathbf{Z}$ else the predictive coding system can choose to skip encoding the innovations in the first step: if the decoder in fact had $\mathbf{Y}'_T = \mathbf{Y}_T$, $\mathbf{X}$ is recovered to within an MSE D without sending anything by setting $\hat{\mathbf{X}} = \mathbf{Y}_T$.) In the high-quality regime, that is, D is very nearly but not quite zero, the percentage rate penalty over R_{distb} given by $(R_{\text{pred}} - R_{\text{distb}})/R_{\text{distb}}$ is very nearly equal to one [22], [23]. In plain words, a predictive coding system requires nearly double the rate (100% penalty) over a distributed coding system at high qualities. Figure 10 shows what this translates to visually in our preliminary implementation of the PRISM distributed video codec [13], [16], [22], [23]. Its operational rate-MSE performance is close to predictive

coding in the synchronized scenario and is substantially better in the mis-synchronized case [22], [23]. Other recent distributed video codec building efforts include [14] and [26].

COMPLEXITY-PERFORMANCE TRADEOFFS

The previous two subsections assumed perfect knowledge of the innovation statistics σ_z^2 (statistics of the correlation between X and $\mathbf{Y}_T$) at both the encoder and the decoder but ambiguity of the motion-index T. Real-world video encoding algorithms involve an "online" learning of the correlation statistics through the process of motion-estimation. Typically, the more the complexity invested in the motion-estimation process the greater the accuracy of the estimates of T and the statistics σ_z^2 leading to better compression performance. However, for video coding over a lossy channel, the *marginal value* of accurately learning the correlation statistics at the encoder diminishes as the channel noise σ_w^2 increases as discussed below. For this discussion, suppose that the component variance of $(\mathbf{Y}_i - \mathbf{Y}_i')$ is roughly the same for all i and is equal to σ_w^2. Suppose the encoder only does a little motion search and settles on the best predictor it has found so far, say $\mathbf{Y}_i$, which may or may not be equal to $\mathbf{Y}_T$. The encoder observes the correlation noise between X and $\mathbf{Y}_i$ to be $\mathbf{Z}_i$ with component variance σ_i^2. As in (2), the rate required by a distributed encoder banking on $\mathbf{Y}_i$ as side information (with a Genie informing the decoder the value of i) is no more than

$$R_{\text{distb}}(D) = \frac{1}{2}\log\frac{\sigma_i^2 + \frac{\sigma_y^2\sigma_w^2}{\sigma_y^2+\sigma_w^2}}{D}.$$

(Note that $\mathbf{Z} = \mathbf{Z}_T$ is roughly independent of all the $\mathbf{Y}_i$s but the same cannot be said of $\mathbf{Z}_i$ for $i \neq T$. If the joint correlation is known, one can potentially use a lower rate.) On the other hand, if the encoder had done full motion search and found $\mathbf{Y}_T$ and $\sigma_T^2 = \sigma_z^2$, the minimum rate required would have been exactly

$$R_{\text{distb}}^{\min}(D) = \frac{1}{2}\log\frac{\sigma_z^2 + \frac{\sigma_y^2\sigma_w^2}{\sigma_y^2+\sigma_w^2}}{D}.$$

The rate reduction obtained from full motion search would be

$$\Delta R = R_{\text{distb}} - R_{\text{distb}}^{\min} = \frac{1}{2}\log\frac{\sigma_i^2 + \frac{\sigma_y^2\sigma_w^2}{\sigma_y^2+\sigma_w^2}}{\sigma_z^2 + \frac{\sigma_y^2\sigma_w^2}{\sigma_y^2+\sigma_w^2}}.$$

It can be verified that $d(\Delta R)/d\sigma_w^2 < 0$ if $\sigma_z^2 < \sigma_i^2$. So, the rate rebate obtained by finding the correlation noise accurately, diminishes as channel noise increases.

To summarize, the main take-away messages of this section are: a) "complexity" can be flexibly distributed among sensor nodes with "no significant performance loss" even in synchronized scenarios, b) predictive source coding is "fundamentally mismatched" to information lossy transmission environments, c) the value of correlation knowledge and motion search diminishes with increasing information loss, and d) distributed source coding is a promising alternative for video coding in wireless environments.

PRISM: A PRACTICAL IMPLEMENTATION

So far we have focused on the analysis of distributed source-coding methods for distributed video compression in a sensor network setting. In this section we describe real-world practical modules that emulate the theoretical and architectural principles discussed above. In this context, we present a brief description of a block-motion block-DCT based implementation of the PRISM [16] video coding system.

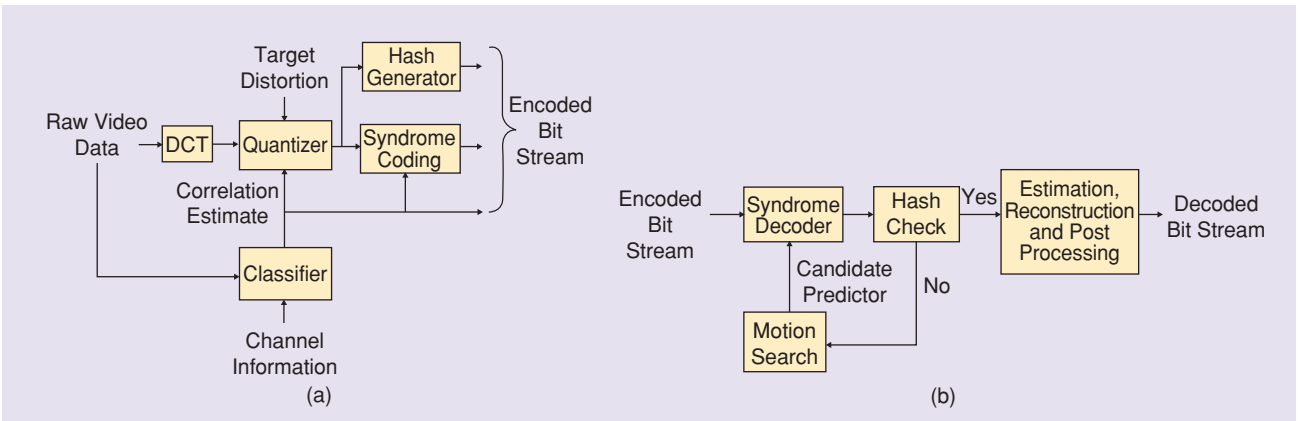
The video frame to be encoded is divided into regular spatial blocks. Using notation similar to previous sections, let X denote the current block to be encoded. Let Y denote the best (motion-compensated) predictor for X in the previous frame and let $\mathbf{X} = \mathbf{Y} + \mathbf{Z}$. We now describe the PRISM encoding and decoding process.

ENCODING

Figure 8(a) shows the encoder block diagram.

CLASSIFICATION

The classification step estimates the correlation noise (Z) of the current video block being encoded. As was discussed earlier, this enables the choice of the appropriate channel code. This



[FIG8] (a) PRISM encoder and (b) PRISM decoder.

step is vital since real video sources exhibit spatiotemporal correlation structures whose statistics are highly spatially varying. Within the same frame, different blocks exhibit different degrees of correlation with their temporal predictors. We use the motion search operation (as in predictive codecs) in combination with offline statistical training to estimate the correlation noise $\mathbf{Z}$. As was discussed, the amount of motion search to be done at the encoder depends upon both the complexity constraints of the encoding and decoding devices as well as the prevailing channel conditions. For instance, the encoder can perform a coarse-level motion search and reveal the outcome to the decoder.

TRANSFORM AND QUANTIZATION

Each block is transformed using the two-dimensional DCT. The transformed coefficients are then scalar quantized with a target quantization step size to come up with the quantized coefficients $\hat{\mathbf{X}}$. The step size is chosen based on the desired reconstruction quality.

SYNDROME ENCODING

As in the illustrative example in "Illustrative Example for Coding with Side Information," the space of quantized code words is partitioned into cosets of a channel code. The number of partitions depends on the strength of the correlation noise $\mathbf{Z}$—the more the correlation (i.e., smaller $\mathbf{Z}$) the less the number of partitions required. Specifically, in our implementation, we partition the quantized code-word space into cosets of a multilevel code [27]. After partitioning the quantized code-word space, the encoder outputs the index of the coset (syndrome) containing $\hat{\mathbf{X}}$.

HASH GENERATOR

While at the encoder, we generate a syndrome for the current block based on estimate of the "best" motion predictor $\mathbf{Y}$ and correlation noise $\mathbf{Z}$, at the decoder, all that is available is the frame memory. As discussed below, the encoder needs to transmit a hash (of sufficient strength) for $\hat{\mathbf{X}}$ in order to facilitate motion estimation at the decoder.

DECODING

Figure 8(b) shows the main decoder modules.

MOTION ESTIMATION AND SYNDROME DECODING

In this framework, the decoder shares the task of motion search with the encoder. As in the SEASON theoretical framework, for each candidate predictor in its list, the decoder performs syndrome decoding to obtain a quantized code word. The decoder then computes the hash of the decoded code word. If the hash of the decoded code word matches the transmitted hash, a success is declared; else the decoder moves on to the next predictor. In case the encoder has performed a partial motion search, the decoder can use that to limit its search list to contain the outcome revealed by the encoder plus the complement of the list used by the encoder.

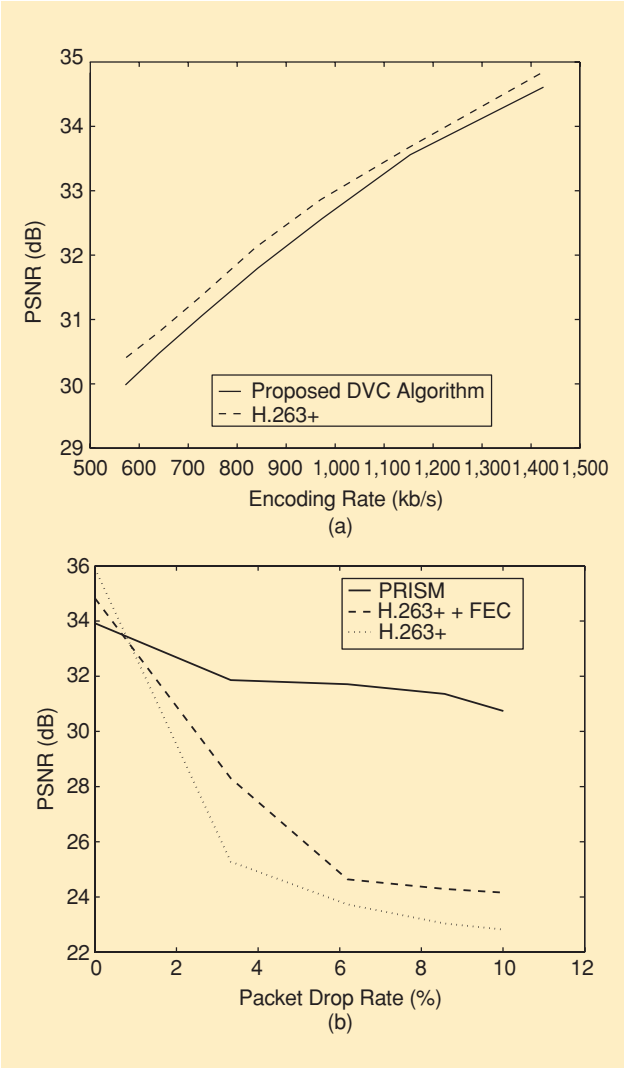
ESTIMATION AND RECONSTRUCTION

Once the quantized code-word sequence is recovered, it is used along with the predictor to obtain the best reconstruction of the source. Any of the sophisticated postprocessing mechanisms can be deployed here to improve the overall performance.

REPRESENTATIVE RESULTS

We now present some experimental results that illustrate the various features of the PRISM video coding framework. Since we use the same coding primitives as the H.263+ [9] video compression standard, we use it as a reference system for our comparisons.

Figure 9(a) compares the compression performance of PRISM and H.263+ for the football (352 × 240, 15f/s) video sequence. Both the systems use the same motion search strategy



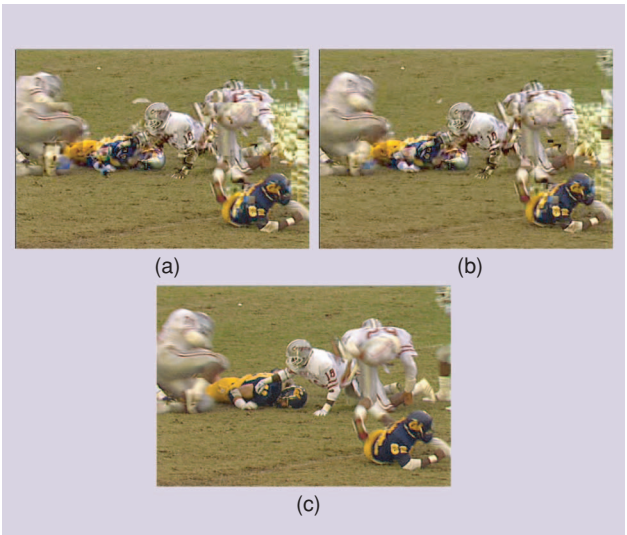
[FIG9] (a) Lossless channel: Comparison of proposed distributed video coding algorithm and H.263+ for the football sequence (352 × 240, 15 f/s) over different encoding rates. (b) Lossy channel: Comparison of PRISM, H.263+ and H.263+ protected with forward error correcting (FEC) codes (Reed-Solomon codes used, 20% of total rate used for parity bits) over a simulated CDMA2000 1X channel for the Football sequence (352 × 240, 15 f/s, 1,700 kb/s).

(full search) at the encoder. As can be seen from Figure 9(a), the performance of the proposed scheme nearly matches that of H.263+. This highlights the fact that distributed source coding based video codecs can approach the performance of prediction based coders when they estimate the correlation structure accurately through the use of good motion models.

We also present the results of some robustness tests conducted on the PRISM system. For these, a wireless channel simulator obtained from Qualcomm Inc. was used. This simulator adds packet errors to multimedia data streams transmitted over wireless networks conforming to the CDMA2000 1X standard [28]. (The packet error rates are determined by computing the carrier to interference ratio of the cellular system.) We tested PRISM, H.263+ and H.263+ protected with FECs (Reed-Solomon codes used, 20% of total rate used for parity bits) over this simulated wireless channel. Here the PRISM system does not do any motion search at the encoder while the H.263+ codec does a full motion search at the encoder. Figure 9(b) shows the performance comparison of these three schemes over a range of error rates for the football (352 × 240, 15 f/s, 1,700 kb/s) sequence. Figure 10 shows the decoded visual quality for the three schemes for the football sequence at 8% average error rate. (Note that 8% is merely the average error rate; the channel is not an independent erasure channel and is in fact quite bursty.) As can be seen in Figure 10, PRISM is able to recover from past errors while error propagation continues to occur for both H.263+ and H.263+ protected with FECs resulting in a better decoded quality. This happens because the FEC-based error resiliency scheme also suffers from drift propagation once the number of packet erasures exceeds the correction capability of the FEC. In contrast, the PRISM scheme is able to correct most of the errors, leading to better decoded video quality.

WIRELESS VIDEO SENSOR NETWORKS ARE CHARACTERIZED BY DEVICES WITH LIMITED PROCESSING CAPABILITIES AND BATTERY POWER CONSTRAINTS, HARSH LOSS-PRONE WIRELESS CHANNELS, AND LOW BANDWIDTHS.

Note that in this setting, PRISM does not do any motion search at the encoder and so loses to H.263+ at 0% loss rate due to inaccurate modeling of the correlation noise statistics. However, as channel noise increases, the importance of such accurate modeling diminishes and the robustness advantages of distributed video coding starts to dominate leading to significant performance gains (over even H.263+ protected with FEC) as highlighted in Figure 9(b).



[FIG10] Decoded visual quality of the ninth frame of the football sequence (352x240, 15 f/s, 1700 kb/s) encoded using (a) H.263+, (b) H.263+ protected with Forward Error Correcting (FEC) codes (Reed-Solomon codes used, 20% of total rate used for parity bits), and (c) PRISM. In each case 15 frames were encoded and then sent over a simulated CDMA2000 1X channel. Note the very annoying drift artifacts in both H.263+ and H.263+ protected with FECs. PRISM has been able to gracefully recover from past errors.

CONCLUDING REMARKS AND FUTURE DIRECTIONS

In this article, motivated by the stringent requirements imposed by the emergence of video sensor networks, we have argued for an architectural paradigm shift in video compression and transmission. Specifically, driven by the need to develop light, robust, energy-efficient, and low delay video delivery schemes, we have described a distributed video coding based framework dubbed PRISM that addresses the wireless video sensor network requirements far more effectively than current state-of-the-art standards like MPEG. We have described the architectural platform, the theoretical foundations, as well as the bridge from theory to video practice, and presented promising experimental evidence based on real-world video sequences that validate the efficacy of our proposed solution.

While our treatment has been primarily confined to a single-camera setup, our proposed paradigm scales naturally to the multiple-camera scenario that we believe will form the cornerstone of emerging video sensor networks (See "Scene Super Resolution Through the Network" for an exciting application of the multiple-camera scenario). The fundamental architectural traits of PRISM, which include robustness, light-encoder architecture, as well as the flexibility in distributing the computational burden of motion estimation between transmitter and receiver, are extremely well suited to the generalization from the single-camera to the multicamera regime. Indeed, as the scale of the network increases in the future, the architectural benefits of PRISM will be magnified. The full potential of large-scale ubiquitous video sensor networks of the future will require an interdisciplinary approach involving signal and video processing, computer vision, multiterminal information theory, and wireless networking. The work presented here represents an important first step towards this goal.

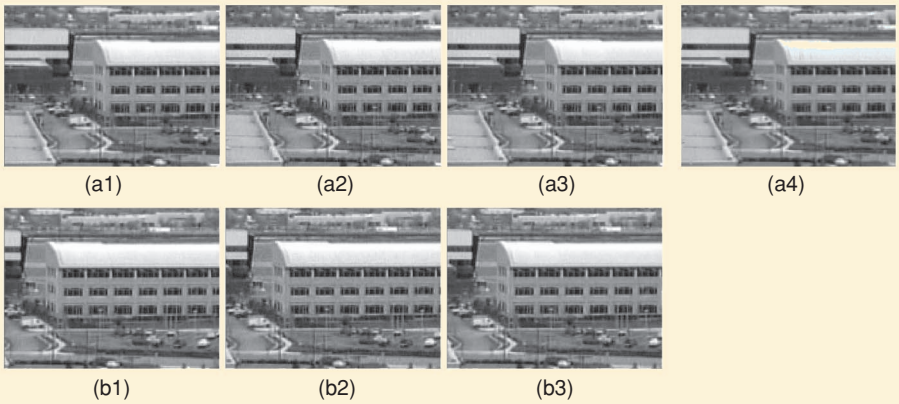
SCENE SUPER-RESOLUTION THROUGH THE NETWORK

Imagine a dense configuration of cameras conducting surveillance in the parking lot of your office building. These cameras have overlapping coverages, and each of these individual cameras is an inexpensive low-resolution device. For instance, each of these cameras can offer a low frame rate (low temporal resolution). An interesting question that arises here is whether all these low-resolution observations can be synergistically combined providing a "virtual super-resolution" system that allows for enhanced capabilities ranging from novel spatiotemporal viewpoint generation/rendering with robustness to individual camera failures? This is indeed feasible, as is demonstrated by Figure 11, which shows three consecutive video frames from two adjacent cameras *A* and *B*. Even though the middle frame in stream *A* (a2) is missing (for example when *A* operates at half the frame-rate of *B*), sophisticated processing based on camera-motion (between *A* and *B*) as well as object-motion modeling enables a near-perfect reconstruction of the missing scene (a4).

Additionally, we can also ask if these correlated data can be efficiently compressed for the purpose of archiving/storage. The increasing relevance of this problem can be gauged from the fact that an industry-wide initiative [5] has been launched recently in the International Standards Organization (ISO) MPEG group with the purpose of addressing this question.

The caveat here is that our sophisticated processing/compression algorithms require all the frames to be present at one central location. While this is easy to resolve in the high-bandwidth wired network case, where the individual cameras can communicate their respective streams (uncompressed or marginally compressed) to the central processing location, this can be a real daunting task in the low-bandwidth, harsh transmission environment wireless network case. It is here that we can use distributed compression algorithms to reduce our transmission bandwidth as well as provide natural robustness to the vagaries of the wireless transmission environment.

We realize that this problem requires an interdisciplinary approach leveraging the latest advances in the areas of signal and video processing, computer vision as well as wireless networking. However, the fundamental architectural features of PRISM, that include robustness as well as ability to share computational complexity between different network nodes, offer the necessary building blocks that form the core of the solution for this problem.



[FIG11] (a) Video stream from camera A including the "original" middle frame (a2) and the reconstructed missing middle frame (a4). (b) Video stream from camera B.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grants CCR-0330514, CCR-0219722, and NSF CAREER CCF-0546598.

AUTHORS

Rohit Puri (rpuri@eecs.berkeley.edu) received the B.Tech. degree from the Indian Institute of Technology, Bombay, the M.S. degree from the University of Illinois at Urbana Champaign, and the Ph.D degree from the University of California, Berkeley, in 1997, 1999, and 2002, respectively, all in electrical engineering. From 2003 to 2004, he was with Sony Electronics Inc., San Jose, and then with the Electrical Engineering and Computer Science Department, University of California, Berkeley. He is currently a senior video architect at PortalPlayer Inc., San Jose. His research interests include multimedia compression, distributed source coding, multiple descriptions coding, and multiuser information theory. He was awarded the Institute Silver Medal by the Indian Institute of Technology, Bombay in 1997. He was a recipient of the 2004 Eliahu I. Jury Award at the University of California, Berkeley.

Abhik Majumdar (abhik@eecs.berkeley.edu) received the B.Tech. degree from the Indian Institute of Technology (IIT), Kharagpur, and M.S. and Ph.D. degrees from the University of California at Berkeley, in 2000, 2003, and 2005, respectively, all in electrical engineering. He is currently with Pure Digital Technologies, San Francisco. His research interests include multimedia compression and networking and wireless communications. He was awarded the 2000 Institute Silver Medal from I.I.T. Kharagpur.

Prakash Ishwar (pi@bu.edu) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Bombay, in 1996, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Urbana-Champaign in 1998 and 2002, respectively. After two years in the Electronics Research Laboratory and the Department of Electrical Engineering and Computer Sciences at the University of California, Berkeley, he joined Boston University where he is an assistant professor in the Department of Electrical and Computer Engineering and a faculty member in the Information Systems and Sciences group, the Center for Information and Systems Engineering, and the Sensor Network Consortium. He was awarded the 2000 Frederic T. and Edith F. Mavis College of Engineering Fellowship of the University of Illinois. He received the 2005 NSF CAREER award. He was the chair of exhibits and demonstrations at the 3rd IEEE/ACM International Symposium on Information Processing in Sensor Networks in 2004 and was coorganizer of Berkeley-FuSe 2003. His research interests include distributed signal processing, information theory, image and video coding, statistical signal processing and modeling, decision theory, multiresolution signal analysis, and optimization theory with applications to sensor networks, multimedia-over-wireless, and information security.

Kannan Ramchandran (kannanr@eecs.berkeley.edu) received the M.S. and Ph.D. degrees from Columbia University in 1984 and 1993, respectively, in electrical engineering. From 1984 to 1990, he was with AT&T Bell Labs. From 1993–1999, he was with the Electrical and Computer Engineering Department at the University of Illinois at Urbana-Champaign and the Beckman Institute and the Coordinated Science Laboratory. Since fall 1999, he has been an associate professor in the Electrical Engineering and Computer Sciences Department, University of California, Berkeley. His current research interests include distributed algorithms for signal processing and communications, multiuser information theory, wavelet theory and multiresolution signal processing, and unified algorithms for multimedia signal processing, communications, and networking. He has received many awards, including the 1993 Eliahu I. Jury Award, the 1997 National Science Foundation (NSF) CAREER award, the 1996 ONR and and 1997 ARO Young Investigator awards, and the 2000 Okawa Foundation Award. In 1998, he was selected as a Henry Magnusky Scholar at the University of Illinois. He is the corecipient of two Best Paper Awards from the IEEE Signal Processing Society, has been a member of the IEEE Image and Multidimensional Signal Processing Committee and the IEEE Multimedia Signal Processing Committee, and was an associate editor for *IEEE Transactions on Image Processing*.

REFERENCES

[1] W.-C. Feng, B. Code, E. Kaiser, M. Shea, W.-C. Feng, and L. Bavoil, "Panoptes: Scalable low-power video sensor networking technologies," in *Proc. 11th ACM Int. Conf. Multimedia*, Berkeley, CA, Nov. 2003, pp. 2–8.

[2] Q. Zhao, A. Swami, and L. Tong, "The interplay between signal processing and networking in sensor networks," *IEEE Signal Processing Mag.,* vol. 23, no. 4, pp. 84–93, 2006.

[3] X. Zhu, A. Aaron, and B. Girod, "Distributed compression for large camera arrays," in *Proc. IEEE Workshop Statistical Signal Processing*, St. Louis, Missouri, Sept. 2003, pp. 30–33.

[4] N. Gehrig and P. L. Dragotti, "Different-distributed and fully flexible image encoders for camera sensor network," in *Proc. Int. Conf. Image Processing*, Genova, Italy, Sept. 2005, vol. II, pp. 690–693.

[5] "Preliminary call for proposals on multi-view video coding," in *Proc. ISO/IEC JTC1/SC29/WG11 N7094*, Busan, Korea, Apr. 2005 [Online]. Available: http://www.chiariglione.org/mpeg/tutorials/technologies/mp-mv/index.htm

[6] D.A. Hazen, R. Puri, and K. Ramchandran, "Multi-camera video resolution enhancement by fusion of spatial disparity and temporal motion fields," in *Proc. IEEE Int. Conf. Computer Vision Systems*, New York City, Jan. 2006, p. 38.

[7] B.G. Haskell, A. Puri, and A.N. Netravali, *Digital Video: An Introduction to MPEG-2.* Norwell, MA: Kluwer, 1996.

[8] S. Battista, F. Casalino, and C. Lande, "MPEG-4: A multimedia standard for the third millennium. 1," *IEEE Multimedia,* vol. 6, no. 4, pp. 74–83, Oct.–Dec. 1999.

[9] G. Cote, B. Erol, M. Gallant, and F. Kossentini, "H.263+: Video coding at low bit rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 7, pp. 849–866, Nov. 1998.

[10] T. Wiegand, G.J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.,* vol. 13, no. 7, pp. 560–576, July 2003.

[11] D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory,* vol. 19, pp. 471–480, July 1973.

[12] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[13] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," in *Proc. 40th Allerton Conf. Communication, Control, and Computing*, Allerton, IL, Oct. 2002.

[14] A. Aaron, R. Zhang, and B. Girod, "Wyner-Ziv coding of motion video," in *Proc. 36th Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 2002, pp. 240–244.

[15] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-Ziv coding of video: An error-resilient compression framework," *IEEE Trans. Multimedia,* vol. 6, no. 2, pp. 249–258, Apr. 2004.

[16] R. Puri, A. Majumdar, and K. Ramchandran, "PRISM: A video coding paradigm based on motion-compensated prediction at the decoder," submitted for publication.

[17] F.J. Macwilliams and N.J.A. Sloane, *The Theory of Error Correcting Codes.* Amsterdam, The Netherlands: Elseiver, 1977.

[18] G.D. Forney, "Coset codes-part I: Introduction and geometrical classification," *IEEE Trans. Inform. Theory,* vol. 34, no. 5, pp. 1123–1151, Sept. 1988.

[19] P. Ishwar, V. Prabhakaran, and K. Ramchandran, "Towards a theory for video coding using distributed compression principles," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, Sept. 2003, vol. II, pp. 687–690.

[20] T.M. Cover and J.A. Thomas, *Elements of Information Theory.* New York: Wiley, 1991.

[21] T. Berger, *Rate Distortion Theory: A Mathematical Basis for Data Compression.* Englewood Cliffs, NJ: Prentice-Hall, 1971.

[22] P. Ishwar, A. Majumdar, R. Puri, and K. Ramchandran, "Analysis of motion-complexity and robustness for video transmission," in *Proc. Wirelesscom,* (special session on distributed video coding), Maui, Hawaii, Jun. 2005, pp. 1261–1265.

[23] A. Majumdar, "PRISM: A video coding paradigm based on source coding with side-information," Ph.D. dissertation, UC, Berkeley, Dec 2005.

[24] A. Aaron, S. Rane, D. Rebollo-Monedero, and B. Girod, "Systematic lossy forward error protection for video waveforms," in *Proc. Int. Conf. Image Processing (ICIP),* 2003, vol. I, pp. 609–612.

[25] A. Majumdar, J. Wang, K. Ramchandran, and H. Garudadri, "Drift reduction in predictive video transmission using a distributed source coded side-channel," in *Proc. ACM Multimedia,* 2004, pp. 404–407.

[26] A. Sehgal, A. Jagmohan, and N. Ahuja, "Wyner-Ziv coding of video: An error-resilient compression framework," *IEEE Trans. Multimedia,* vol. 6, no. 2, pp. 249–258, Apr. 2004.

[27] A. Majumdar, J. Chou, and K. Ramchandran, "Robust distributed video compression based on multilevel coset codes," in *Proc. Asilomar,* Nov. 2003, pp. 845–849.

[28] *TIA/EIA Interim Standard for CDMA2000 Spread Spectrum Systems*, May 2002.

SP