THE INSTITUTE FOR SYSTEMS RESEARCH

ISR TECHNICAL REPORT 2010-9

Energy and Thermal-Aware Video Coding via Encoder/Decoder Workload Balancing

Domenic Forte, Ankur Srivastava

Energy and Thermal-Aware Video Coding via Encoder/Decoder Workload Balancing

Domenic Forte
University of Maryland
Dept. of Electrical and Computer Engineering
College Park, MD 20742
dforte@umd.edu

Ankur Srivastava
University of Maryland
Dept. of Electrical and Computer Engineering
College Park, MD 20742
ankurs@umd.edu

ABSTRACT

Even with consistent advances in storage and transmission capacity, video coding and compression are essential components of multimedia services. Traditional video coding paradigms result in excessive computation at either the encoder or decoder. However, several recent papers have proposed a hybrid PVC/DVC (Predictive/ Distributed Video Coding) codec which shares the video coding workload. In this paper, we propose a controller for such hybrid coders that considers energy and temperature to dynamically split the coding workload of a system comprised of one encoder and one decoder. Results show that the proposed controller results in more balanced energy utilization, improving overall system lifetime and reducing operating temperatures when compared to strictly PVC and DVC systems.

Keywords

thermal management, energy management, Wyner-Ziv coding, Distributed Video Coding, video coding

List of Abbreviations			
PVC	Predictive Video Coding	SI	Side Information
DVC	Distributed Video Coding	MV	Motion Vector
ME	Motion Estimation	MB	Macroblock
TE	Turbo Encoding	GOP	Group-of-Picture
TD	Turbo Decoding		

1. INTRODUCTION

1.1 Motivation

Even with consistent advances in video storage and transmission capacity, video coding and compression are essential components of multimedia services [1]. They allow for digital video use in environments that could not support raw video and make more efficient use of transmission and storage resources. The most conventional video coding technique is Predictive Video Coding (PVC) where the video encoder exploits interframe correlations to achieve high compression efficiency. PVC was designed to benefit applications with down-link models, such as digital television broadcasting, where a sender encodes a video signal once and that signal is decoded by many receivers. In this model, the encoder has enough energy and compute power while the decoder is resource constrained. However, with the emergence of wireless networks, there are a growing number of applications which follow an up-link model and require light encoding or a flexible distribution video coding.

At the turn of the century, new research focusing on a fundamentally different coding paradigm called Distributed Video Coding (DVC) shifted workload to the decoder, leading to a low complexity encoder [2]. The theory essentially says that if the correlation between frames in a video sequence is known at the decoder, DVC can achieve the same rate-distortion performance as PVC [3]. Since the correlation for video sequences is never explicitly known, the encoder must still play a minor role in the coding process by sending error correcting information to the decoder. In this model, the decoder performs the majority of the workload.

One of the emerging challenges in video streaming scenarios is coping with applications (eg. heterogeneous devices) that don't conform to strict down-link or uplink models. For example, a video being transferred between two cellphones. Unfortunately, neither PVC nor DVC can provide a flexible distribution of coding workload where resources available at the sender and receiver are considered throughout video transfer. This paper is concerned with dynamically balancing complexity of a hybrid PVC/DVC system consisting of a single encoder and single decoder separated by a communication medium to satisfy resource constraints at both ends of the system. In this work, we have chosen to jointly optimize energy-related and thermal resources. For video coding and transfer applications, the available energy in battery constrained devices changes and strictly conforming to a DVC or PVC model may be inefficient for the coding duration. Furthermore, if even a static policy for energy does indeed hold, safe operating temperatures might not be guaranteed throughout video transfer.

1.2 Related Work

There has been a great deal of work on energy/power management in PVC coding, specifically MPEG-x and H.26x standards. A nice overview of power-aware work for mobile multimedia and video coding is presented in [4]. Techniques discussed include transform approximations, motion estimation search heuristics, and scalable (layered) video coding. In [5], the authors develop a power-rate-distortion model for an MPEG encoder with complexity control parameters and solve an optimization problem to maximize video quality given power and framerate constraints.

[6, 7] both proposed dynamic thermal management schemes for MPEG decoding by allowing a certain amount of image/video quality loss. Each work made use of dynamic voltage/frequency scaling and some graceful performance degradation policies to ensure a thermally safe state of operation. To our knowledge, there has been no work on thermal management schemes for the MPEG encoding in a real-time system which is surprising since MPEG encoding requires 5-10 times the complexity of decoding [2].

Flexible workload sharing between encoder and decoder through hybrid PVC/DVC was shown in [8, 9]. In [8], the encoder performed interframe prediction only for portions of a video frame with high levels of motion activity. The decoder side was responsible for predicting the remainder of the frame and combining results with the encoder's information. Experiments showed that increasing encoder participation improved rate-distortion performance. In [9], interframe prediction was divided between the encoder/decoder in two ways: spatial partitioning (similar to [8]) and temporal refinement. The authors modeled prediction complexity for both ends of the system and solved an optimization problem to satisfy fixed encoder/decoder complexity constraints. In our literature survey, we could not find any papers dealing explicitly with temperature or energy for DVC or hybrid

PVC/DVC coders.

1.3 Objective and Contributions

Our main contribution is an autonomous and adaptive video coding controller that distributes workload between a sender (encoder) and receiver (decoder) in order to simultaneously balance energy consumption and meet manufacturer specified thermal constraints at both sides of the system. The autonomous and adaptability criteria are of critical importance because the statistical characteristics of the video frames and the tradeoffs associated with their compression and transmission vary over time. While the complexity resource problem presented in [9] can certainly extend to energy in some sense, their work is far more general. Additionally, our problem considers thermally safe temperatures across encoder/decoder boundaries which to our knowledge has not been covered in any video coding related paper. Research in traditional video coding techniques only focuses on the excessive computing and/or transmission required by one end of the system. To solve the dynamic workload balancing problem, we develop models to estimate energy consumed, power, temperature, and data transmission and apply them to a specific problem instance. Results obtained for five benchmark videos show improvements in overall system lifetime (20% on average) when comparing the proposed method to a strictly PVC coding system. Regarding temperature, results show a reduction in thermal violations at encoder and decoder when comparing strict PVC and DVC systems to the proposed.

2. VIDEO CODING OVERVIEW

In this section, we give a high level overview of PVC, DVC, and a PVC/DVC hybrid codec. We make the common assumptions that a video frame F is decomposed into $N \times N$ pixel non-overlapping macroblocks (MB) which undergo motion estimation, transformation (typically DCT), and quantization to remove spatial and temporal redundancy and increase compression efficiency. A simple understanding of motion estimation (ME) for calculating motion vectors (MVs) by the reader is assumed as well.

2.1 Predictive Video Coding (PVC)

In Predictive Video Coding such as interframe prediction in MPEG standards, block-based ME is performed entirely by the encoder using reference frames also available at the decoder. Following ME, information is entropy encoded and sent to the decoder and used to reconstruct the frame. The resulting frame may not be identical to the original video frame due to bit errors from channel and the lossy transformation/quantization process performed for additional compression. For more on the MPEG/H.264 process, see [1].

2.2 Distributed Video Coding (DVC)

In Distributed Video Coding, ME is performed entirely by the decoder. The most popular DVC architecture is the Stanford Wyner-Ziv transform domain coding architecture (see [2] for finer details) where the video sequence is split into key (I) frames and Wyner-Ziv (W) frames. In short, I frames are encoded and decoded using an intraframe prediction/compensation such as H.264/AVC Intra and inserted periodically determining the Group-of-Picture (GOP) size.

At the encoder, W frames are essentially transformed (typically DCT), quantized, and turbo encoded (TE). At the decoder, side information (SI) is generated to form a noisy estimate W' for W. Using SI, the turbo decoder (TD) requests parity information from the encoder and computes iteratively until a stopping criterion is met. More accurate SI means the stopping criterion is easier to achieve and less parity information is needed at the decoder. Following the successful TD, the frame is reconstructed and inversely transformed.

2.3 Hybrid Video Codec

For applications that cannot fully follow either the up-link or down-link coding paradigms, the use of resources in PVC and DVC is unsatisfactory and only provides two extremes. In this paper, our goal is to provide a flexible system comprised of one encoder and one decoder that can dynamically adapt to resource constraints at both sides. The resources we are concerned with are energy and temperature.

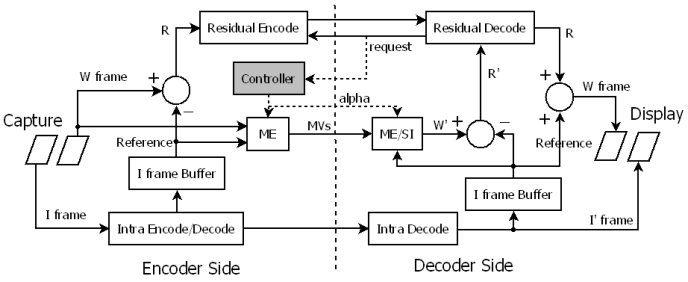

Figure 1: Hybrid Video Codec with controller (filled block) placed at encoder side

In video coding, a group of pictures, or GOP structure, specifies the order in which intra- and inter-frames are arranged. "Intra-coding" refers to the various lossless and lossy compression techniques performed relative to spatial information contained only within the current frame, and not relative to any other frame in the video sequence. "Inter-coding" exploits the similarities among successive frames and achieves superior compression to "Intra" at the cost of increasing computational and storage requirements. Every coded video stream consists of successive GOPs.

The proposed system uses the PVC/DVC hybrid codec (i.e. '**co**der-**dec**oder'.) coarsely shown in Figure 1 and follows the Stanford DVC architecture with residual coding [10]. For simplicity, we have grouped transform, quantization, and turbo coding modules into "residual encode" and "residual decode" blocks. The hybrid codec's GOP structure is as follows. Each incoming frame F is designated as an I frame (for intra-coding) or W frame (for inter-coding) depending on GOP length l for inserting I frames. For example, the GOP for $l = 2$ and $l = 3$ would be $\{I, W\}$ and $\{I, W, W\}$ respectively. In the hybrid codec, I frames are encoded/decoded with the H.264 Intra codec and are used as references for preceding and next W frames. For W frames, rather than compressing and transmitting W, a reference frame I_{ref} is used to improve coding efficiency by computing residual $R = W - I_{\text{ref}}$. The more similar W is to I_{ref}, the less energy R contains.

Typically, inter-coding compression at the encoder (PVC) and decoder (DVC) is accomplished with expensive block-based motion estimation (ME) algorithms. In PVC, the encoder transmits ME produced motion vectors (MVs) and residual R so the decoder can reconstruct W. For DVC, ME at the decoder side results in an estimate of the W frame, W', also referred to as side information (SI). After computing $R' = W' - I_{\text{ref}}$ (I_{ref} has been sent by the encoder and stored in a buffer), the turbo decoder (TD) iteratively computes and requests only enough of R to correct R' and reconstruct W with some confidence. The amount of R requested depends on how well W' approximates W (i.e. how accurate SI is). In this work, we will assume that initially c bits for error correction are always transmitted and additional bits are sent to the decoder upon request from the TD.

Unlike strict PVC and DVC, the hybrid codec is flexible and ME can be performed at the encoder, decoder, or a combination of the two [8, 9]. In our case, power and temperature at the encoder and decoder are dependent and can be balanced by controlling how ME is distributed. The resource usage depends on how much ME is performed at either side and how much data is requested by the TD. Our main contribution to the hybrid codec is a controller which maintains resource status of the entire system and dynamically splits ME through parameter α. At extremes $\alpha = 0$ and $\alpha = 1$, the system is strictly a DVC and PVC system respectively. By increasing/decreasing α, the ME complexity increases at the encoder/decoder sides. Aside from balancing ME complexity, sharing the ME process should also result in tradeoffs in the TD module.

3. RESOURCE MODELS AND PROBLEM

3.1 Problem Statement

Consider the hybrid video coding system described in Figure 1 with GOP $l = 2$ and some initial energies at the encoder and decoder sides. Our goal is to dynamically obtain an α^* that maximizes the lifetime of the system while maintaining operating temperatures $T_{\text{enc}}(t)$ and $T_{\text{dec}}(t)$ below thermal constraint C_T. In the rest of this section, we develop our resource models and formulate the joint optimization problem.

3.2 Energy Models

In this section, we discuss our energy consumption models for the encoder and decoder. For some modules in Figure 1, the energy required depends on the video sequence and/or channel conditions. However, we assume these are constants because our models can easily be extended when the frame and/or channel statistics are known. In this paper, we strive for more accuracy on modules dependent on α which is our main contribution. α is the workload balancing parameter where extremes $\alpha = 0$ and $\alpha = 1$ result in strictly DVC and PVC systems respectively. Note that at both sides of the system, we will assume the transmission/reception energy needed for **data requests** is negligible.

3.2.1 Encoder

The energy consumption model for W frames at the encoder is given by

$$E_{\text{enc}}(\alpha, W) = \begin{cases} E_{\text{R,e}} + f_e(\alpha)E_{\text{ME,e}} + \left(\frac{d(\alpha)+c}{c}\right) E_{\text{Tx},W} + E_{\text{MV,e}} & , \alpha > 0 \\ E_{\text{R,e}} + \left(\frac{d(\alpha)+c}{c}\right) E_{\text{Tx},W} & , \alpha = 0 \end{cases} \quad (1)$$

$E_{\text{R,e}}$ accounts for "residual encoding" energy. $f_e(\alpha)$ is an implementation specific function residing in the range [0,1] that scales the energy required to perform full ME at the encoder, $E_{\text{ME,e}}$. For residual bit transmission, $E_{\text{Tx},W}$ is the energy spent in transmitting c bits. In this work, we assume that a fixed number of bits, c, are always transmitted and $d(\alpha)$ additional bits may be requested depending on the accuracy of SI at the decoder. Hence, $d(\alpha)$ represents the amount of residual bits that need to be transmitted. In section 4, we will discuss a simple prediction method for $d(\alpha)$ that depends on the recent bit transmission history. $E_{\text{MV,e}}$ is the energy required to encode and transmit MV information. We assume that only the MV accuracy depends on α and the number of MVs sent to the decoder is the same unless $\alpha = 0$. When $\alpha = 0$, the encoder does not perform ME so total energy consumed does not include $E_{\text{ME,e}}$ or $E_{\text{MV,e}}$. For I frames, we assume the energy required to intra encode, decode, and transmit the compressed information is constant $E_{\text{enc},I}$.

3.2.2 Decoder

The energy model for W frames at the decoder follows similarly and is given by

$$E_{\text{dec}}(\alpha, W) = \begin{cases} E_{\text{R,d}} + f_{\text{d}}(\alpha)E_{\text{ME,d}} + \left(\frac{d(\alpha)+c}{c}\right)(E_{\text{TD}} + E_{\text{Rx}}) \\ \quad +E_{\text{MV,d}} & , \alpha > 0 \\ E_{\text{R,d}} + E_{\text{ME,d}} + \left(\frac{d(0)+c}{c}\right)(E_{\text{TD}} + E_{\text{Rx}}) & , \alpha = 0 \end{cases} \quad (2)$$

$E_{\text{R,d}}$ refers to the energy needed for all processes in the "residual decoding" module **except TD**. $f_{\text{d}}(\alpha)$ scales $E_{\text{ME,d}}$, the energy for the decoder's full ME/SI process. On this side, $\left(\frac{d(\alpha)+c}{c}\right)$ is used to scale the amount of energy for receiving residual bits, E_{Rx}, as well as the energy needed to TD c bits, E_{TD}. $E_{\text{MV,d}}$ refers to the energy spent in receiving and entropy decoding MVs. Like the encoder side, $E_{\text{MV,d}}$ is only required when the encoder performs ME ($\alpha > 0$) and has MVs to send. For I frames, we assume the energy required to receive compressed information and intra decode is constant $E_{\text{dec},I}$.

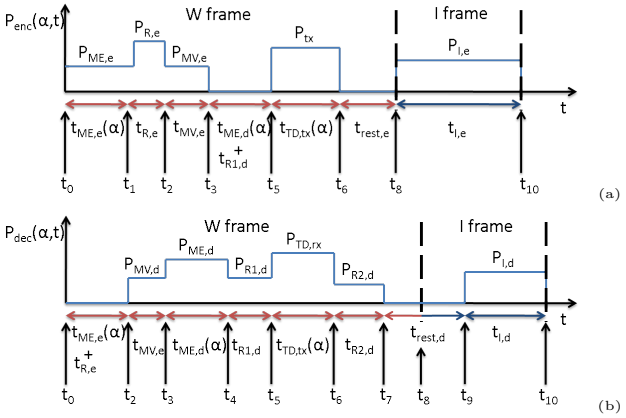

Figure 2: (a)Encoder and (b)decoder side power profiles for (W,I) pairs w.r.t. α. *Please note both are for illustrative purposes and are not drawn to scale.*

3.3 RC Thermal Model

We model the encoder/decoder hardware as processors with communication hardware. The thermal behavior of such electronic systems can be modeled by an RC circuit with voltage representing the temperature and current representing power dissipation. The resistance R is the potential heat path throughout the system, while capacitance C indicates the ability of the system to store heat [11].

The RC thermal model consists of the following relationship between the core temperature/ambient temperature potential difference T and power consumption $P(t)$:

$$\frac{dT}{dt} = -\frac{T}{RC} + \frac{P(t)}{C} \quad (3)$$

In this work, we will model encoder and decoder hardware with a single R and C parameter each. For improved accuracy, one could model the processor and communication hardware separately with two coupled RC models. This can be covered by future works. The role of this paper is to model thermal behavior of the physically separate, but dependent encoder and decoder.

The power profiles we use in the thermal evaluation of a {W,I} pair for GOP length 2 and $\alpha > 0$ are shown in Figure 2. We assume for simplicity that processing for W and I frames can be finish within time durations to achieve visually acceptable frame rates. At the encoder side (Figure 2(a)), W frames begin with the ME process which takes a time duration dependent on α, $t_{\text{ME,e}}(\alpha)$, and requires constant power consumption $P_{\text{ME,e}}$. Next, the residual encoding and MV encoding/transmission occur with constant power consumptions and within constant time durations. The encoder can rest in the next interval which takes $t_{\text{ME,d}}(\alpha)$ time units during which the decoder is performing ME, generating SI, and performing the portion of residual decoding required for TD. Within this interval, we ignore static leakage at the encoder and assume zero power consumption for simplicity. Following rest, the encoder sends at least c residual bits to the decoder. For simplicity, we assume a constant power level for transmission,P_{Tx}, over this interval which lasts for $t_{TD,tx}(\alpha)$ units of time. $t_{TD,tx}(\alpha)$ is the time it takes to transmit $d(\alpha)$, the amount of data exchanged as a function of α. The encoder rests for another constant time period where the decoder finishing the residual decoding modules and computing $W = R + F_r$. For I frames, we assume a constant power level $P_{\text{enc},I}$ over the entire $t_{\text{e},I}$ time interval for simplicity. The decoder power profile (Figure 2(b)) is explained in a similar fashion and the power profiles for $\alpha = 0$ (not shown) are only slightly modified versions of these so we omit the details for brevity.

The salient point to take from the power profiles is how the time durations for ME and TD depend on α. Altering α will clearly have an effect on thermal behavior because it shrinks or enlarges the time spent (and energy consumed overall) in several modules. For process states that require constant nonzero power P_x beginning at initial time t_i and ending at

final time t_f, we solve Equation (3) and find that:

$$T(t) = T_{nat} + (T(t_i) - T_{\text{nat},i})e^{-(t-t_i)/RC} \quad , t_i \leq t \leq t_f \tag{4}$$

$$\text{where } T_{\text{nat},i} = RP_i \tag{5}$$

Additionally, for process states with zero power (neglecting static leakage) beginning at initial time t_i and ending at final time t_f, one may solve Equation (3) for:

$$T(t) = T(t_i)e^{-(t-t_i)/RC} \quad , t_i \leq t \leq t_f \tag{6}$$

Given a power profile $P(t)$, such as either in Figure 2, one can apply Equations (4) and (6) to model system thermal behavior.

3.4 Joint Optimization Problem

Consider the hybrid video codec system described in Figure 1 along with the energy and thermal models previously described. Assuming energy to process an $\{I, W\}$ frame pair at the encoder and decoder is $E_{\text{enc}}(\alpha)$ and $E_{\text{dec}}(\alpha)$, our goal is to maximize the lifetime of the entire system by choosing

$$\begin{aligned} \alpha^* &= \text{argmin}\left(\max(E_{\text{enc}}(\alpha), \beta E_{\text{dec}}(\alpha))\right) \\ &\text{s.t. } 0 \leq \alpha \leq 1 \\ &T_{\text{enc}}(t), T_{\text{dec}}(t) < C_T \; \forall \, t \end{aligned} \tag{7}$$

for each GOP in the video sequence. Intuitively, we design a controller to find an α for every GOP that minimizes the energy spent by the harder working coder side while maintaining thermal constraints at both sides. Weighting factor β is used to favor the encoder or decoder (if initial energies available were not the same for example). Otherwise, $\beta = 1$. The objective captures the fact that the system will be non-functional if either the encoder or decoder fails.

Note that the complexity of the optimization problem defined by Equation (7) depends critically on $d(\alpha)$, $f_{\text{e}}(\alpha)$, and $f_{\text{d}}(\alpha)$ which we will explore in the next two sections. We provide an analytical solution to the problem in Section 6.

4. CONTROLLER DESIGN

The controller is located at the encoder side of the system (see Figure 1). Its main function is to solve the optimization problem by using the energy models, power profiles, and thermal models discussed above. Although the controller knows how much energy will be consumed for the majority of modules at the encoder and decoder for any choice of α, it is impossible to know for sure how much residual data $d(\alpha)$ will be requested because $d(\alpha)$ depends on the random statistical characteristics of the video frame sequence. Therefore, before solving the optimization problem, the controller must predict $d(\alpha)$.

Frames in a video sequence have different levels of motion activity and correlation with reference frames. Therefore, predicting the additional bit exchange, $d(\alpha)$, is challenging as it varies from frame to frame. We assume that $d(\alpha)$ is similar for several consecutive W frames since motion activity is typically correlated temporally. In our implementation, we use an offline training phase and online history to form a simple prediction of $d(\alpha)$.

In the offline training phase, we calculate $d(\alpha)$ on a per frame basis for some discrete points in α. Training frames are randomly selected from benchmark video sequences to characterize $d(\alpha)$ for different frame sequences. The $d(\alpha)$s from training frames are fitted by n-degree polynomials and collectively form a library D. Ideally, D is representative of all levels of motion activity and correlation with reference frames. The fit for $d(\alpha)$ for a single training frame and $n = 2, 6$ is shown in Figure 3. We found that our fits using experimental data were monotonically decreasing as α increases which makes intuitive sense. The SI's accuracy should be greater when the encoder is more involved in ME. From this point forward, we will **only consider linear fits($\mathbf{n = 1}$) for $\mathbf{d(\alpha)}$** to simplify analysis.

In D we store a set of typical $d(\alpha)$ functions and dynamically choose a particular $d(\alpha)$ for a given set of frames in real-time. Specifically, the controller maintains a bit transmission history buffer d_{hist} which holds how much data has been recently requested for a given α^*. To estimate how much

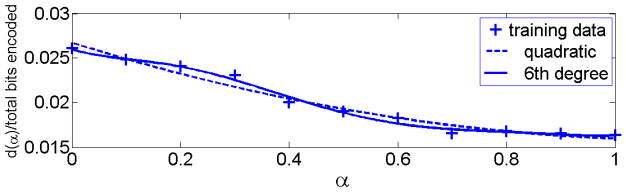

Figure 3: Polynomial fits for $d(\alpha)$ of a training frame.

data will be exchanged, the controller chooses a fit from library D with closest resemblance to d_{hist}. With an estimate of data exchange, the controller may now solve the optimization problem for α^* (see Eq. (7)).

Rather than solve for every GOP in a video sequence, our controller maintains the same choice for α^* for several consecutive W frames. We use an heuristic method to decide when a new α^* is needed. When the variance in d_{hist} exceeds a fixed threshold, we assume that the previous prediction for data exchange no longer holds and restart the above process. Analytical methods for α^* depend on the nature of $d(\alpha)$, $f_e(\alpha)$, etc. and will be discussed in Section 6. The next section discusses a specific implementation.

5. PROBLEM INSTANCE: RASTER SCAN-BASED ME

This section highlights how the ME process is divided between the encoder and decoder. We use block-based ME algorithm at both sides of the system with fixed parameters macroblock (MB) length N and search window length S. We hope to justify approximations for $f_e(\alpha)$ and $f_d(\alpha)$ from Equations in Section 3.2 as well as $t_{ME,e}(\alpha)$ and $t_{ME,d}(\alpha)$ from Figure 2.

In block-based ME, a frame F is divided into $N \times N$ pixel non-overlapping MBs. Each MB in F is predicted from an MB of equal size in reference frame F_{ref}. MBs are not transformed in any way apart from being shifted to the position of the predicted MB. This shift is identified by a motion vector(MV). Let F_X denote an MB centered at pixel X. For F_X, Figure 4(a) shows several possible MBs and their respective motion vectors (MVs) in F_{ref}. Our ME is performed for all MBs with a full-search algorithm in raster scan order. In full search, F_X is compared to every possible MB within $F_{\text{ref},X}$, an $S \times S$ search window centered at pixel X in F_{ref}. Raster scan order (Figure 4(b)) implies that the search proceeds left to right in the window's first row, then left to right in the next, and so on. The MB in $F_{\text{ref},X}$ that results in the lowest mean square error (MSE) is chosen as the predicted MB and its corresponding MV is recorded.

In our implementation, the encoder performs forward and backward ME between current frame W and the immediately preceding and next I frames (references). Parameter α increases and decreases the area scanned in the search window. For $0 < \alpha < 1$, only a portion on the raster scan is performed (Figure 4(c)). Therefore, we use $f_e(\alpha) = \alpha$ in our energy model and $t_{ME,e}(\alpha) = \alpha t_{ME,e}$ for the encoder's power profile. Note that $t_{ME,e}$ is a constant time duration needed to perform full search ME when $\alpha = 1$.

At the decoder side, we follow the method in [12] for creating predicted MBs. In short, forward and backward ME is performed between the preceding and next I frames since the W frame doesn't reside at the decoder. Then SI is generated by linearly interpolating the resulting MVs. Once again, the parameter α alters the search. However, for this side, the search proceeds in a reverse raster scan order (Figure 4(d)) to test the range omitted by the encoder ME process. Therefore, we use $f_d(\alpha) = 1 - \alpha$ to scale the energy used at the decoder for full ME. We ignore the SI portion of the procedure which involves at worst an averaging of pixel values for overlapping MBs [12].

6. JOINT OPTIMIZATION SOLUTION

Given our implementation's $f_e(\alpha)$, $f_d(\alpha)$, $d(\alpha)$ (note that among these $d(\alpha)$ is being predicted dynamically by the controller), and all the respective time durations (such as $t_{e,I}$,$t_{ME,e}$,

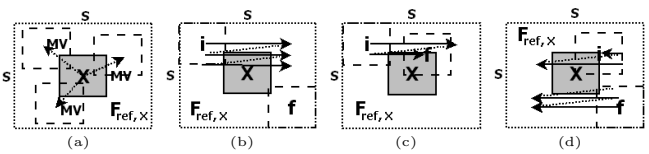

Figure 4: (a) MVs (arrows) for several possible MBs (dashed boxes) in $F_{ref,X}$. The gray block represents the MB F_X being predicted. (b) Full MB search in raster scan order. For our hybrid codec with $\alpha \approx .28$, the partial raster scans are shown for (c) encoder and (d) decoder. i and f denote the initial and final MBs in any search.

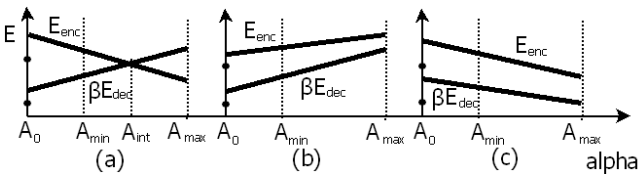

Figure 5: Candidate α's possible for 3 cases to determine $\alpha^* = \text{argmin}\,(\max(E_{\mathbf{enc}}(\alpha), \beta E_{\mathbf{dec}}(\alpha)))$

etc.), we know the power profiles and energy spent for any choice of α. The solution to our joint optimization problem follows a general methodology. First, we obtain a range of α that can successfully maintain $T_{\text{enc}}(t), T_{\text{dec}}(t) < C_T$. Then we restrict the energy minimization problem to α within that range. Assuming the system can physically satisfy thermal constraints, this method should result in an optimal solution.

6.1 Identifying Feasible Workload Ranges

In order to obtain a feasible range for α where thermal constraints are not violated, we need

$$A_{\text{enc}} = \{0 \le \alpha_i \le 1 : T_{\text{enc}}(t, \alpha_i) < C_T \; \forall \, t\} \quad (8)$$

$$A_{\text{dec}} = \{0 \le \alpha_i \le 1 : T_{\text{dec}}(t, \alpha_i) < C_T \; \forall \, t\} \quad (9)$$

Rather than explicitly calculating $T(t) \; \forall \; t$ or several GOPs, we offer a less tedious approach and find one maximum temperature per side. If the maximum temperature for a given α does not violate C_T then clearly no other temperature can violate. Assuming a known $d(\alpha)$, the power's periodic behavior (Fig. 2) will eventually result in a repeating thermal pattern (or "stable state") with identical period as $t \to \infty$. From this point forward, we will refer to this identical period as the thermal period. In short, this stable state occurs when the temperatures at the start and end (t_0 and t_{10} in Fig. 2) of a GOP are equal ($T(t_0) = T(t_{10})$) for both the encoder and decoder sides.

We are interested in the thermal period's maximum temperature at each side of the system and will use it to find feasible ranges for α. By evaluating temperature within the stable state, the problem is simplified because a maximum temperature may only occur at certain time instances within the thermal period. The temperature is guaranteed to be monotonically decreasing over intervals with zero power consumption (Eqn. (6)) which removes them from consideration. For those non-zero power intervals where Eqn. (4) holds, the temperature may monotonically increase or decrease depending on the nature of the stable state. However, a maximum can **only occur at the end of one of these intervals** (see Fig. 2). Therefore, we find the maximum temperature for both encoder and decoder by comparing the temperatures at only these points:

$$T_{\max,\text{enc}}(\alpha) = \max(\{T(t_1), T(t_2), T(t_3), T(t_6), T(t_{10})\}) \quad (10)$$

$$T_{\max,\text{dec}}(\alpha) = \max(\{T(t_3), T(t_4), T(t_5), T(t_6), T(t_7), T(t_{10})\}) \quad (11)$$

For $T_{\max,\text{enc}}(\alpha)$, we solve a linear system of equations so that we can obtain general expressions for each temperature in Eqn. (10) in terms of α. For the encoder, there are seven equations:

$$T(t_0) = T(t_{10}) = T_{\text{nat},8} + (T(t_8) - T_{\text{nat},8})e^{-t_{I,\text{e}}/RC} \quad (12)$$

$$T(t_1) = T_{\text{nat},0} + (T(t_0) - T_{\text{nat},0})e^{-t_{\text{ME,e}}(\alpha)/RC} \quad (13)$$

$$T(t_2) = T_{\text{nat},1} + (T(t_1) - T_{\text{nat},1})e^{-t_{\text{R,e}}/RC} \quad (14)$$

$$T(t_3) = T_{\text{nat},2} + (T(t_2) - T_{\text{nat},2})e^{-t_{\text{MV,e}}/RC} \quad (15)$$

$$T(t_5) = T(t_3)e^{-(t_{\text{ME,d}} + t_{\text{R1,d}})/RC} \quad (16)$$

$$T(t_6) = T_{\text{nat},5} + (T(t_5) - T_{\text{nat},5})e^{-t_{\text{TD,tx}}(\alpha)/RC} \quad (17)$$

$$T(t_8) = T(t_6)e^{-t_{\text{rest,e}}/RC} \quad (18)$$

where $T_{\text{nat},i} \forall i$, $t_{I,\text{e}}$, ME,e(α), etc. are known parameters for any given α. There are eight equations for the decoder side which are generated similarly and will not be shown for brevity.

With the above, we can find the maximum temperature in the steady state at both sides of the system for any given α. However, we must also find all α satisfying the thermal constraints. Recall that $f_e(\alpha)$ and $f_d(\alpha)$ from Section 5 as well as $d(\alpha)$ from Section 4 are all linear in α. This implies that maximum temperatures are **strictly decreasing or increasing** functions w.r.t. α. The search procedure for α satisfying thermal constraints is performed as follows

Encoder's Range: When the maximum temperature at the encoder is strictly decreasing, we perform a binary search on α and find

$$\alpha_{\text{enc}_\text{l}} \text{ s.t. } \alpha_i \ge \alpha_{\text{enc}_\text{l}}, \, T_{\max,\text{enc}}(\alpha_i) \le C_T$$
$$\alpha_i < \alpha_{\text{x}_\text{l}}, \, T_{\max,\text{enc}}(\alpha_i) > C_T \quad (19)$$

$$A_{\text{enc}} = \{\alpha_{\text{enc}_\text{l}} \le \alpha_i \le 1\} \quad (20)$$

Alternatively for the strictly increasing case, we perform a binary search on α and find

$$\alpha_{\text{enc}_\text{u}} \text{ s.t. } \alpha_i > \alpha_{\text{enc}_\text{u}}, \, T_{\max,\text{enc}}(\alpha_i) > C_T$$
$$\alpha_i \le \alpha_{\text{enc}_\text{u}}, \, T_{\max,\text{enc}}(\alpha_i) \le C_T \quad (21)$$

$$A_{\text{enc}} = \{0 < \alpha_i \le \alpha_{\text{enc}_\text{u}}\} \quad (22)$$

Decoder's range: This range is determined just as the encoder's. Therefore, for A_{dec}, simply replace the "enc" subscript in all of the above with "dec".

α= 0 Case: The encoder and decoder range equations discussed above are subject to $\alpha_i > 0$ because energy consumption follows a different model for $\alpha = 0$ (see Section 3.2). Although the temperatures for $\alpha = 0^+$ may fail to meet thermal constraints, those for $\alpha = 0$ can pass because the encoder doesn't perform ME. This removes the additional burden present at both ends of the system for coding, transmission, and reception of encoder MVs. We perform a separate test with the appropriate models for $\alpha = 0$ considering both the encoder and decoder sides to obtain

$$A_0 = \begin{cases} 0 & , T_{\max,\text{enc}}(0), T_{\max,\text{dec}}(0) < C_T \\ \emptyset & , \text{otherwise} \end{cases} \quad (23)$$

6.2 Energy Optimization Solution

In this section we provide an analytical solution for (7) with lone constraint $\alpha \in \{A_0, A_\cap\}$ where

$$A_\cap = A_{\text{enc}} \cap A_{\text{dec}} \quad (24)$$

$A_\cap$ represents the range of α (excluding 0) that satisfies thermal constraints at both encoder and decoder. If $\{A_0, A_\cap\} = \emptyset$, there is no solution satisfying the thermal constraints. This case could dealt with by some heuristic graceful degradation policies which trade off temperature with video quality and will not be analyzed further in this paper.

For $\{A_0, A_\cap\} \neq \emptyset$, there are only four possibilities for α^*. Clearly, E_{enc} and E_{dec} are **strictly increasing or decreasing linear** functions of α because $f_e(\alpha)$, $f_d(\alpha)$, and $d(\alpha)$ are linear. Therefore, E_{enc} and E_{dec} will intersect at most once. Let us define such a point of intersection

$$A_{\text{int}} = \begin{cases} \alpha_i & , E_{\text{enc}}(\alpha_i) = \beta E_{\text{dec}}(\alpha_i) \text{ and } \alpha_i \in A_\cap \\ \emptyset & , \text{otherwise} \end{cases} \quad (25)$$

Video	PVC			DVC			proposed		
	$T_{\max,\text{enc}}$	$T_{\max,\text{dec}}$	$E_{\max}$	$T_{\max,\text{enc}}$	$T_{\max,\text{dec}}$	$E_{\max}$	$T_{\max,\text{enc}}$	$T_{\max,\text{dec}}$	$E_{\max}$
salesman	73.50*	65.08	1132	65.00	65.25	897	65.27	65.24	898
akiyo	73.50*	65.08	1132	65.00	65.24	897	65.27	65.22	898
carphone	73.58*	65.29	1133	65.00	79.30*	930	69.32	68.47	947
foreman	73.54*	65.14	1133	65.00	78.43*	899	69.48	69.04	936
mobile	73.59*	65.36	1134	65.00	74.81*	944	69.53	71.45*	968

Table 1: Benchmark video results where * denotes a violation of thermal constraint $C_T = 70^o$C.

where β was defined previously in Section 3.4. Let us also define $A_{\min} = \min(\alpha)$ and $A_{\max} = \max(\alpha)$ s.t. $\alpha \in A_\cap$.

Several possibilities are shown in Figure 5 for clarification. Although Figure 5 does not exhaustively show every possibility, one can infer that α^* clearly $\in \{A_0, A_{\min}, A_{\text{int}}, A_{\max}\}$. For simplicity, we test all four cases and the controller selects

$$\alpha^* = \text{argmin}\left(\max(E_{\text{enc}}(\alpha), \beta E_{\text{dec}}(\alpha))\right) \quad (26)$$
$$\text{s.t. } \alpha \in \{A_0, A_{\min}, A_{\text{int}}, A_{\max}\}$$

The above says that optimal workload sharing only occurs at α equal to zero, a point of intersection within $A_\cap$ (if it exists), the lower bound of $A_\cap$, or the upper bound of $A_\cap$.

In summary, the controller selects α^* to maximize the system lifetime and maintain thermally safe operating temperatures. However, the resulting temperatures may still exceed thermal thresholds if $d(\alpha^*)$ is under-estimated or poorly fit. Additionally, the system would also require some graceful degradation policies for cases when it is simply not possible to meet thermal constraints for any choice of α (i.e. $\{A_0, A_\cap\} = \emptyset$).

7. SIMULATION RESULTS

We tested our PVC/DVC hybrid codec and controller using five QCIF benchmark video sequences (salesman, akiyo, carphone, foreman, mobile) with varying levels of motion activity. Experiments were conducted assuming a fixed thermal constraint $C_T = 70^o$C and with initial encoder/decoder temperatures 65^oC. MB length and ME search range were fixed at $N = 8$ and $S = 13$ respectively. The fit library D was composed of 20 linear fits from the salesman and foreman videos. An α^* was chosen whenever the variance in data exchanged exceeded 1%. Before history could be accumulated, $\alpha^* = .5$ was a simple compromise to initialize the system. If α^* could not be found to maintain constraints, α^* was found by solving the energy minimization problem alone. Table 1 shows energy and maximum temperature results for the benchmark videos using strictly PVC ($\alpha = 1$), strictly DVC ($\alpha = 0$), and our proposed hybrid PVC/DVC systems. $T_{\max,\text{enc}}$ and $T_{\max,\text{dec}}$ denote the maximum temperature at the encoder and decoder respectively through simulating 300 frames per video sequence. $E_{\max}$ is the total energy used at the harder working end of the system.

Our hybrid PVC/DVC system shows a better compromise between energy and temperatures for all videos when compared to the stricter codecs. For low motion videos (salesman,akiyo), we see that the strict DVC system is already the optimal solution so our proposed hybrid defaults to DVC behavior after brief initialization. Results for high motion videos (carphone, mobile, foreman) are more interesting. The strict PVC and DVC systems violate C_T at the encoder and decoder sides by approximately 3.5^oC and 7.5^oC on average respectively. On the other hand, the proposed system is able to balance the workload effectively so that C_T is only violated at the decoder for the mobile video and to a lesser extent (1.5^oC). Concerning energy, the PVC system always consumes the most energy at one side of the system. Strict DVC consumes the least energy of any approach, but violates C_T more severely to do so. Considering all videos, the proposed system uses about 20 % less energy on average than strict PVC, but only 1% more energy on average than strict DVC.

8. CONCLUSION AND FINAL REMARKS

The hybrid PVC/DVC system with controller shows promise and with more advanced forms of prediction and graceful degradation, results should be even better. We plan on making other improvements to the energy and thermal models for improved accuracy as well. For instance, we would like to study a thermal model where the processor and communication hardware are coupled and require different thermal constraints. Rather than evaluating the temperature at infinity, we would like to explore how the current temperature at each end of the system can be used for better results. Finally, we want to extend the power models to include static leakage power and study its effects on thermal behavior.

9. REFERENCES

[1] Iain E. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding for Next Generation Multimedia*, Wiley, 1 edition, August 2003.

[2] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[3] Aaron D. Wyner and Jacob Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, pp. 1–10, 1976.

[4] Wu D. Ci S. Wang H. Katsaggelos A. Zhang, J., "Power-aware mobile multimedia: a survey (invited paper)," *Journal of Communications*, vol. 4, no. 9, 2009.

[5] Zhihai He, Yongfang Liang, Lulin Chen, I. Ahmad, and Dapeng Wu, "Power-rate-distortion analysis for wireless video communication under energy constraints," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 15, no. 5, pp. 645–658, May 2005.

[6] Inchoon Yeo and Eun Jung Kim, "Hybrid dynamic thermal management based on statistical characteristics of multimedia applications," in *ISLPED '08: Proc. Intl. Symp. on Low Power Electronics and Design*, New York, NY, USA, 2008, pp. 321–326, ACM.

[7] Wonbok Lee, Kimish Patel, and Massoud Pedram, "Dynamic thermal management for mpeg-2 decoding," in *ISLPED '06: Proc. Intl. Symp. on Low Power Electronics and Design*, New York, NY, USA, 2006, pp. 316–321, ACM.

[8] Hu Chen and E. Steinbach, "Flexible distribution of computational complexity between the encoder and the decoder in distributed video coding," in *Multimedia and Expo, 2008 IEEE International Conference on*, Hannover, June/Apr. 2008, pp. 801–804.

[9] Jrgen Slowack, Jozef Skorupa, Stefaan Mys, Peter Lambert, Christos Grecos, and Rik Van de Walle, "Flexible distribution of complexity by hybrid predictive-distributed video coding," *Signal Processing: Image Communication*, vol. 25, no. 2, pp. 94 – 110, 2010.

[10] Anne Aaron, David Varodayan, and Bernd Girod, "Wyner-ziv residual coding of video," in *Proc. Picture Coding Symp., PCS-2006*, 2006.

[11] Aviad Cohen, Finkelstein, Avi Mendelson, Ronny Ronen, and Dmitry Rudoy, "On estimating optimal performance of cpu dynamic thermal management," *IEEE Comp. Archit. Letters*, vol. 2, no. 1, 2003.

[12] Bruno Macchiavello, Fernanda Brandi, Eduardo Peixoto, Ricardo L. de Queiroz, and Debargha Mukherjee, "Side-information generation for temporally and spatially scalable wyner-ziv codecs," *J. Image Video Process.*, vol. 2009, pp. 1–11, 2009.