ON COMPRESSION OF ENCRYPTED IMAGES

Prof. Ja-Ling Wu

Dept. CSIE &GINM

National Taiwan University

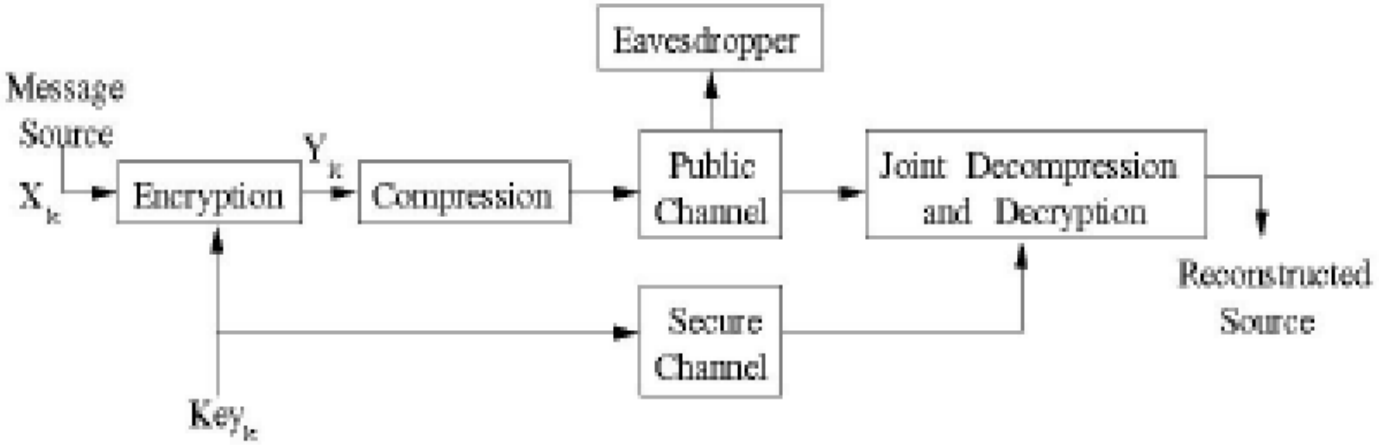
- The content of this lecture is taken from the paper : On Compression of Encrypted Images,
- IEEE International Conference on Images, 2006
- Co-authored by : Daniel Schonberg, Stark Draper, Kannan Ramchandran (U.C. Berkeley)

- In IEEE Transaction on Signal Processing, 2004, the authors shown that it is theoretically possible to compress encrypted data to the entropy rate of the unencrypted source.
- Since good encryption makes a source look completely random, traditional algorithms are unable to compress encrypted data.
- For this reason, traditional systems make sure to compress before they encrypt.

- The above-mentioned paper showed that the problem of compressing encrypted data is related to distributed source coding with side information.
- It was shown that neither compression performance nor security need be compromised under some specific (structural and/or statistical) conditions.

A block diagram of this system structure is in Fig. 1.

- **Fig.1 :**The source is first encrypted and then compressed. The compressor does not have access to the key used in the encryption step. At the decoder, decompression and decryption are performed jointly.



- Fig. 2. A I00xI00 sample binary image is on the left (10, 000 bits).To encrypt this image, the 10, 000 bit random key in the center is added to the unencrypted image on the left.

- In Figure 2, the world map image on the left and the key in center are added via bitwise exclusive-OR to produce the encrypted image (one-time pad encryption) on the right. Though several algorithms exist today for compressing the highly structured unencrypted image on the left, no image compression algorithms exist that can compress the marginally random image on the right.

- To understand why the image on the right is compressible, note that while the unencrypted plain-text and ciphertext are independent, compression is achieved by leveraging the dependence between the cipher-text and the key.
- This dependence can be understood by viewing the cipher-text as a noisy version of the key stream.

- Since the key stream (behaves like the side-information) is available at the decoder, we can reconstruct the cipher text with the compressed data.
- Reconstruction is achieved via Slepian- Wolf coding. In order to develop systems that can compress encrypted data, we develop distributed source coding schemes whose inter source correlations match the unencrypted source's statistics.

- Up to now, practical schemes for compressing encrypted data have focused on simple source models, such as memoryless models.
- There is still a significant gap between these models and "real world" images, which are better modeled by their natural 2-D structure. As can be seen with Fig. 2, a 1-D model is insufficient to capture all the structure in the image.

- In this work, the authors describe how to decode using a model designed to capture the underlying 2-D structure of images. The resultant algorithm is more efficient for compressing encrypted images. A practical scheme, based on LDPC (Low Density Parity Check) codes (for compressing encrypted images) is implemented. The authors also describe how to apply their scheme to binary images.

2. SOURCE MODEL

- In this section, the model for spatially correlated sources is discussed. Let's consider binary images, wherein each pixel $x_{i,j}$ takes on one of two values, i.e., $X_{i,j} \in \{0, 1\}$. Images are sampled on a rectangular grid with N_h rows and N_v columns.
- Traditionally, the image bits were raster scanned and only the correlation between successive bits in the scan was considered.

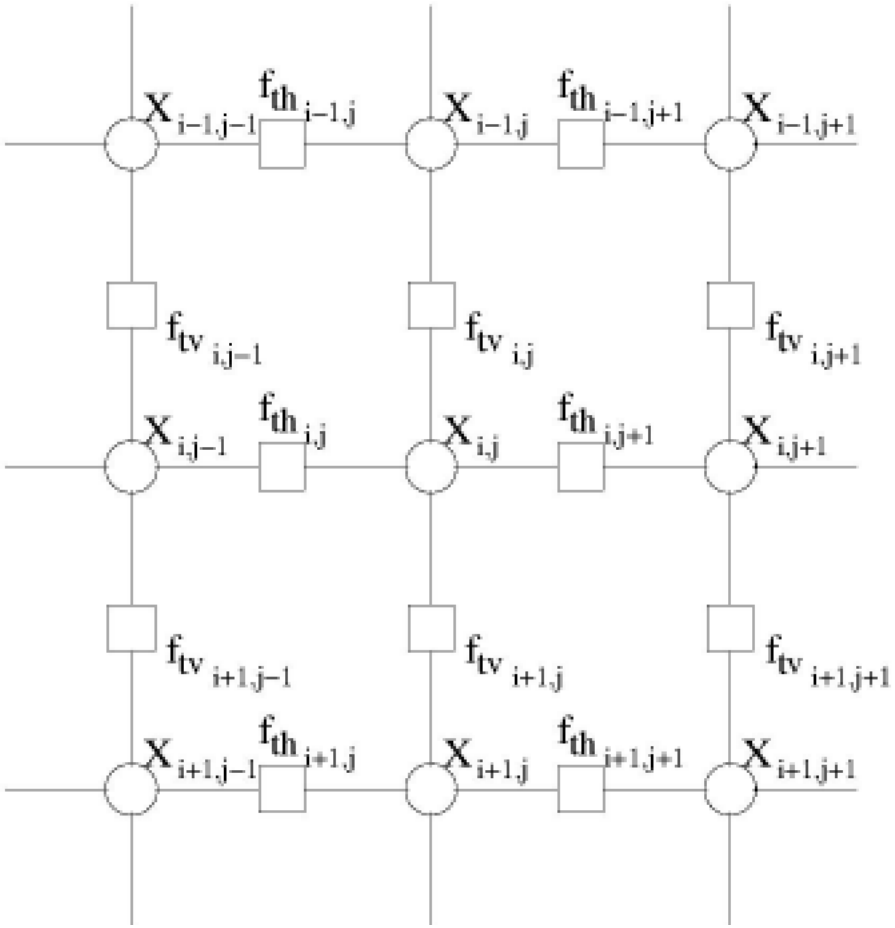
- This forces a 1-D model on the data.
- In this lecture, let's consider the correlation between each pixel and its 4 nearest neighbors; up & down, left &right.
- We consider here a Markov field model for spatially dependent sources instead of a Markov chain.

- The Markov field model is illustrated by way of factor graphs [1]. Factor graphs are bipartite graphs consisting of variables (represented by circles) and constraints on those variables (represented by squares).

- [1] F. Kschischang, B. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," in IEEE Trans. Inform. Theory, Feb. 2001, vol. 47, pp. 498-519.

- A section of the factor graph for the 2-D Markov field model for binary images is presented in Fig. 3.
- In the graph in Fig. 3, the circles labeled $\mathbf{x}_{i,j}$ represent the bits of the image and the squares labeled $\mathbf{fth}_{i,j}$ and $\mathbf{ftc}_{,i,j}$ represent the dependence between pixels.

Fig. 3. A factor graph for the spatial source (Markov field) model.



- We denote the marginal probability on each bit as p = Pr($x_{i,j} = 1$). We take the correlations to be symmetric for both the horizontal and vertical dimensions.
- The horizontal parameters are denoted

$$h_0 = Pr(x_{i,j} = 1 | x_{i,j-1} = 0) = Pr(x_{i,j} = 1 | x_{i,j+1} = 0)$$

and

$$h_1 = Pr(x_{i,j} = 1 | x_{i,j-1} = 1) = Pr(x_{i,j} = 1 | x_{i,j+1} = 0)$$

- Vertical parameters are denoted

$$v_0 = Pr(x_{i,j} = 1 | x_{i-1,j} = 0) = Pr(x_{i,j} = 1 | x_{i+1,j} = 0)$$

and

$$v_1 = Pr(x_{i,j} = 1 | x_{i-1,j} = 1) = Pr(x_{i,j} = 1 | x_{i+1,j} = 1).$$

- Though relatively simple, this model allows us to take into account spatial correlations between a greater number of pixels exists, the nearest neighbor Markov model captures the strongest inter-pixel correlations. As we shall see, the result is a significant performance improvements over the 1-D model.

3. ENCODER AND DECODER

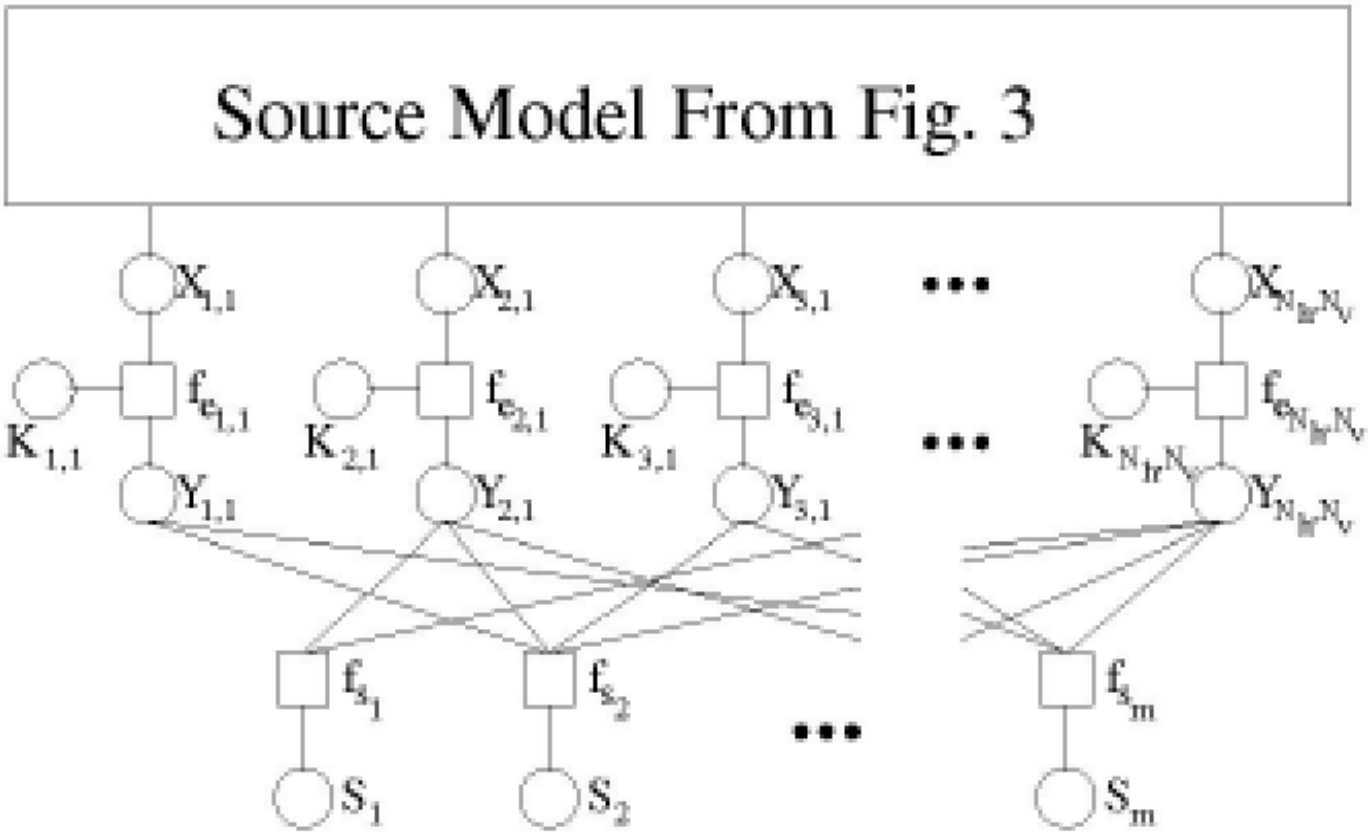
- We begin by assuming that full knowledge of the source statistics (p, h_0, h_1, v_0, v_1) is available to both encoder and decoder, and then relax this assumption later.
- We compress the encrypted source using a sparse linear transformation implemented with a matrix multiplication.

- A detailed description of the design of the linear transformation matrix (and the basis for this codec) can be found in [2]. In particular, the design of the transform matrix is based (with a modification discussed below) on LDPC codes [3].
- [2] D. Schonberg, K. Ramchandran, and S. S. Pradhan, "LDPC codes can approach the Slepian Wolf bound for general binary sources," in 40th Annual Allerton Conf, Oct. 2002, pp. 576-585.
- [3] R. G. Gallager, Low Density Parity Check Codes, Ph.D. thesis, MIT, Cambridge, MA, 1963.

- The decoder operates by running belief propagation over the factor graph. The graphical model consists of three components connected together; the models for the source, the encryption, and the code. Details of the source graphical model were described in Section 2 and shown in Fig. 3.

- We form the encryption model and attach it to the source model as shown in Fig. 4.
- Since we consider only stream ciphers here, we can model the encryption process as $Y_{i,j} = x_{i,j} \oplus k_{ij}$, where $Y_{i,j}$ is the cipher-text, k_{ij} is the bits of the key, and $\oplus$ indicates the exclusive-OR operation.

- Fig. 4. The full graphical model for compressing encrypted spatially correlated sources.
- The model consists of the source model on top (abstracted here but shown in detail in Fig. 3), the encryption model in the middle, and the code on the bottom.


Source Model From Fig. 3
X1,1
X2,1
X3,1
XNIrNV
fe1,1
fe2,1
fe3,1
feNIrNV
K1,1
K2,1
K3,1
KNIrNV
Y1,1
Y2,1
Y3,1
YNIrNV
fs1
fs2
fsm
S1
S2
Sm

- We represent the constraint between these three variables in the graphical model with a square node labeled $\mathbf{fe}_{i,j}$.
- The circles representing the variables $x_{i,j}$, $k_{i,j}$, and $Y_{i,j}$ are all connected to the encryption constraint $\mathbf{fe}_{i,j}$.

- The code model consists of a representation of the linear transformation matrix H, the cipher bits $Y_{i,j}$, and the compressed bits $\mathbf{s}_i$. In [2] it was shown that good performance can be achieved when the transformation matrix is designed as an LDPC code.

- This structure is represented graphically in Fig. 4.
- The squares labeled $\mathbf{fs}_i$, represent the linear transformation H, and the results of that transformation are represented by the circles labeled $\mathbf{s}_i$ (i.e., the compressed bits).

- Decoding is achieved using the sum-product algorithm on the factor graph of Fig. 4. The sum-product algorithm is an inference algorithm designed to be exact on trees.
- Although not exact on "loopy" graphs (such as the graph in Fig. 4), empirical performance is very good.

- Strong performance is due both to the code sparsity (thus its loops are long on average) and the source being smooth (thus there is strong dependency between adjacent bits). The algorithm iteratively updates an estimate of the distribution for each of the variables.

- In the first half of each iteration, the constraints (squares) update their messages while in the second half of each iteration, the variables (circles) respond by updating their messages.
- Messages represent the current distribution estimate.

- In the sparse linear transformation a portion of the output bits are designed to be exactly equal to the source bits. We refer to these as "doped" (hidden) bits. The doped bits are our modification of the LDPC code based design. Typically between 30% and 50% of the compressed bits are doped bits.

- These bits are used in two ways. First, since these doped bits are known unambiguously at the decoder they anchor the iterative decoding process by catalyzing the process. Second, they provide a mechanism for estimating the statistics of the masked (encrypted) source.

- By selecting the doped bits to come in adjacent pairs, the decoder can empirically estimate the source statistics and use the estimates for decoding.
- To relax the assumption that the encoder also know the source statistics, feedback could be used by the decoder after estimating the source statistics [4].

- [4] D. Schonberg, S. Draper, and K. Ramchandran, "On blind compression of encrypted correlated data approaching the source entropy rate," in 43rd Annual Allerton Conf:, Allerton, IL, Sep. 2005.

4. RESULTS

- As a demonstration, we compress the encrypted version of the binary image leftmost in Fig. 2. The unencrypted 100x100 binary image (10, 000 bits) is a binary map of the globe. We use the doped bits in order to calculate the source statistics for use in the sum-product algorithm run at the decoder. This image is encrypted (right image in Fig. 2) by adding a pseudorandom Bernoulli-1/2 string (center image in Fig. 2).

- In this example, the presented method compresses the encrypted data to 4, 299 bits, of which 2, 019 are doped bits.
- The decoder empirically estimates (p, h_0, h_1, v_0, v_1) = (0.3935, 0.0594, 0.9132, 0.0420, 0.9295) and then reconstructs the original image using 81 iterations.

- The compressed bits and the reconstruction are presented in Fig. 5.
- For comparison, we present the 1-D memory model used in [4], where the encrypted image could only be compressed to 7, 710 bits. In that simulation, the image was reconstructed in 27 iterations. The 2-D source model allows for greater compressibility.

- Fig. 5. A comparison of the compressed bits and reconstructed image using the 1-D memoryless model from [4] and the 2-D memory model presented here. The 1-D model compressed the encrypted data to 7, 710,

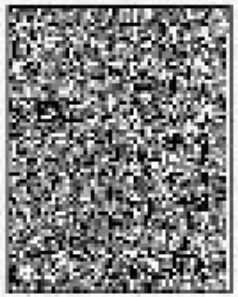
  3, 411 more bits than the 4, 299 bits used for

  the 2-D model.

  Clearly, the 2-D model achieves greater

  compression.

COMPRESSION
RATE = 0.77



DECODE
& DECRYPT

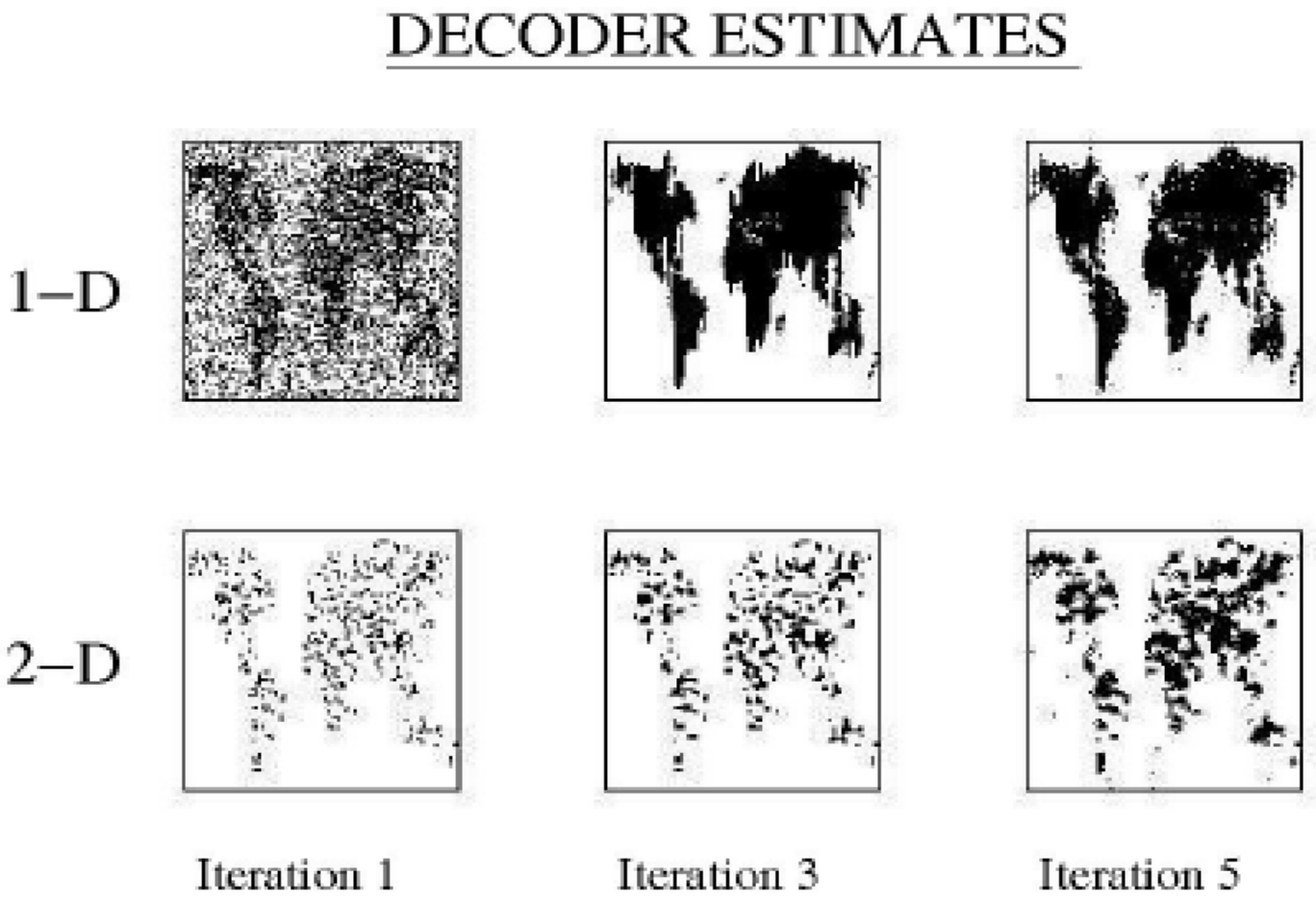
COMPRESSION
RATE = 0.43

DECODE
& DECRYPT

- To see the effects of the source model on the decoding, we present the estimates of the 2 decoders at the end of three iterations in Fig. 6. The 1-D decoder estimates can be seen to exhibit artifacts resulting from the north-south raster scanning.
- These artifacts are the several visible up and down lines.

- In contrast, such artifacts do not show up with the 2-D decoder.
- Instead, the estimates seem to result in "clumped" areas, areas that grow from iteration to iteration. For both decoders, after a handful of iterations (typically under 10), these artifacts disappear.

Fig. 6


DECODER ESTIMATES
1–D
2–D
Iteration 1
Iteration 3
Iteration 5

5. CONCLUSIONS & FUTURE DIRECTIONS

- This work naturally suggests an extension to gray-scale and other larger-alphabet (color) images. A first approach is to break an image up into a series of bit-planes where each bit-plane represents all the bits of equal significance in the binary expansion of the pixel values.

- Image structure is typically highly concentrated in the most significant bit-planes though. As a result, little compression gain is available with this approach.
- Accurate image models are necessary to be able to achieve significant gains when compressing encrypted data.

- Modeling images presents several new challenges though. Most existing image compression algorithms are based on transforms, e.g., the DCT (Discrete Cosine Transform) in JPEG. Transforms aim to convert the image into a domain where it may be represented with only a few coefficients.

- Using a bitwise stream cipher though, it becomes impossible to consider transforms since encryption is a non-linear process.
- In contrast, image coders which use pixel domain models use highly non-stationary predictors. For example, JPEG-LS (lossless) compresses each pixel based on 4 of the adjacent pixels.

- Since this data is unavailable when the image is encrypted, application is not straightforward.
- Instead, compression of encrypted gray-scale image will require greater use of the doped bits and other learning techniques.

- By contrast, encrypted video offers advantages unavailable to single encrypted images. As an example, consider 3 frames of a video.
- At the decoder, after the first 2 frames are decoded, the third frame can be estimated from the two previous frames with high reliability by considering "motion" models.

- Since the difference between this estimate and the actual frame is likely to be small, compression gains could be significant. Temporal dependence (as with most popular video coders) may offer greater compressibility than the spatial dependence considered in this lecture, and is a promising area of future study.

- IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 3, NO. 4, DECEMBER 2008
- Toward Compression of Encrypted Images and Video Sequences
- Daniel Schonberg, Stark C. Draper, Chuohao Yeo, *and* Kannan Ramchandran