Chromatic Encoding: a Low Power Encoding Technique for Digital Visual Interface

Wei-Chung Cheng and **Massoud Pedram**
Dept. of EE Systems
University of Southern California
Los Angeles CA

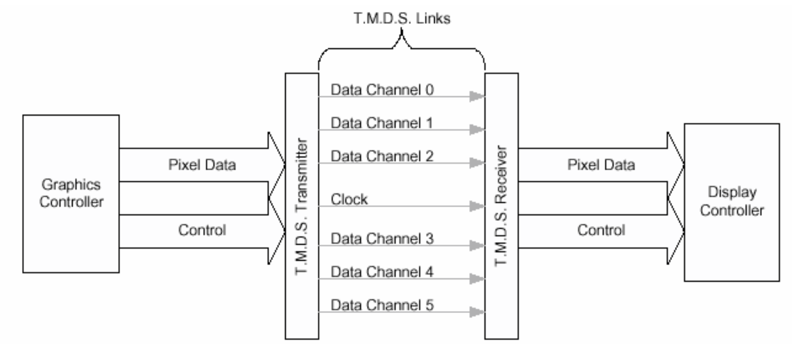
March 6, 2003

Outline

- Digital Video Interface Standard
- Tonal Locality
  - Concept
  - Validation
- Encoding Framework
  - Spatial Encoding
  - Chromatic Encoding
  - Ordered Transition Codes
- Experimental Results
- Conclusion

Target System and DVI

- Video Interface Classification
  - Analog (VGA)
  - Digital
    - Parallel (LCD)
    - Serial (DVI)
  - Multiple color channels
- Digital Visual Interface (DVI)
  - Digital + Serial + Multiple channels



Digital Video Interface

- Previous Digital Interfaces
  - VESA Plug and Display, Digital Flat Panel, OpenLDI; Not widely accepted
- Digital Visual Interface (DVI)
  - Defined by Digital Display Working Group (DDWG)
  - Sponsored by Intel, IBM, Compaq, NEC, HP, Fujitsu
  - Derived from PanelLink from Silicon Image; Version 1.0 released in April 1999
- Transition Minimized Differential Signaling (TMDS)
  - Current-mode differential signaling
- Encoding is required to
  - Reduce transitions, correct errors, reduce the DC current, and minimize EMI

TMDS Encoding

- TMDS encoding function
  - Maps the 8-bit original pixel value *D* into a 10-bit encoded data value *E*
    - *E*[9..0]=TMDS(D[7..0])
  - *E*[8]: shows whether XOR or XNOR operation is used
    - G[0]=D[0]
    - For 0 < k < 8 // to reduce the intra-word transitions
      if E[8] == 0 then G[k] = D[k] XNOR G[k-1]
      else G[k] = D[k] XOR G[k-1]
  - *E*[9]: shows whether inversion or copy operation is used
    - For 0 <= k < 8 // to reduce DC current
      if E[9] == 0 then E[k] = G[k]
      else E[k] = ~G[k]
- Key drawback
  - TMDS encoder does not make use of the image characteristics

DVI Transition Count Function

- For each pixel *i* in the frame-buffer, its 8-bit signal value x_i is encoded as the 10-bit $e(x_i)$ and sent serially through the corresponding channel. The sum of the transition counts in a DVI data channel is:

$$\sum_i \left(\sum_{j=0}^{8} e(x_i)[j] \oplus e(x_i)[j+1] + e(x_i)[9] \oplus e(x_{i+1})[0] \right)$$

- The first term represents the *intra-word* transitions whereas the second term represents the *inter-word* transitions. We ignore the inter-word transition count, which is much smaller.
- The goal is to find an optimal code assignment that minimizes:

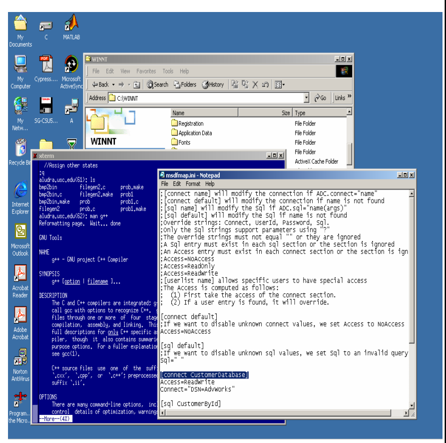
$$\sum_i w(e(x_i)) \text{ where } w(x) = \sum_{j=0}^{8} x[j] \oplus x[j+1]$$

Tonal Locality

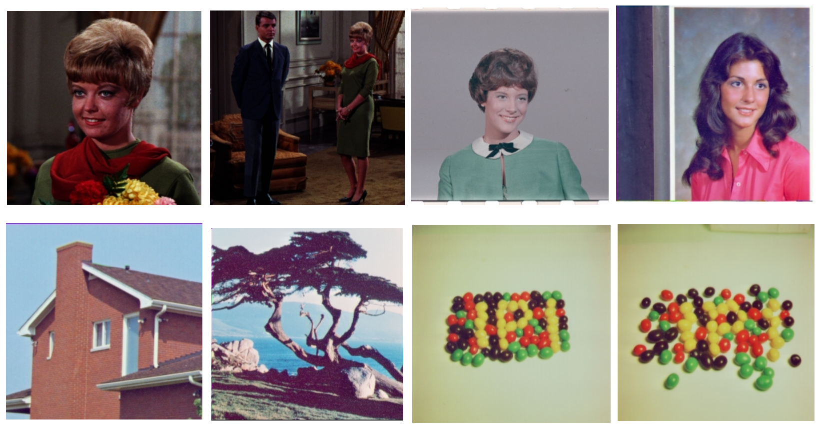
- Tonal locality is the notion that a pixel is likely to be similar to its neighboring pixels in terms of the signal values. Furthermore, the signal differences between adjacent pixels follow a Gaussian distribution
- Sources
  - Objects
    - Continuous Texture
  - Light Tracing
    - Reflection
    - Diffusion
  - Optical Systems
    - Lens resolution
    - Recording Media
  - Artifacts
    - Compression

Tonal Locality and Edge Detection

- Edge detection is the basis of many image processing algorithms
- Procedure
  - Big difference across the boundary -> edge
  - Not detected as an edge -> small difference
- Text-mode image
  - Exhibits an even higher degree of tonal locality



USC SIPI Image Database



- USC SIPI Image Database, Vol. 3 Miscellaneous, 4.1.01~4.1.08
- Size: 256*256
- Color Depth: 24-bit, [0,255] per R, G, B
- TIFF format

Spatial Correlation Between Adjacent Pixels

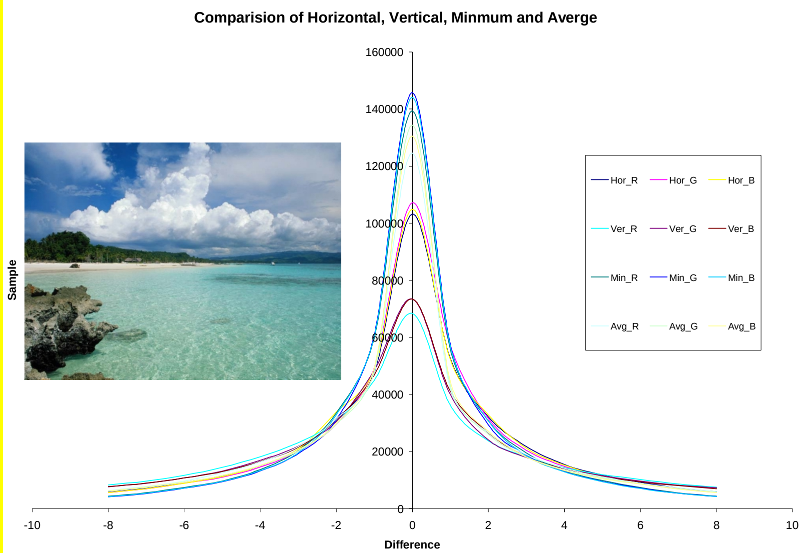
- **Coefficients of Determination**
- **Correlation between neighboring pixels**
- **Compares pixel(i,j) with pixel(i+m,j+n)**
- **R G B separately**

R	*i*	*i+1*	*i+2*	*i+3*	*i+4*
j	1.00	0.97	0.93	0.88	0.84
j+1	0.96	0.95	0.91	0.87	0.83
j+2	0.90	0.90	0.87	0.84	0.81
j+3	0.85	0.84	0.83	0.81	0.78
j+4	0.79	0.79	0.79	0.77	0.75

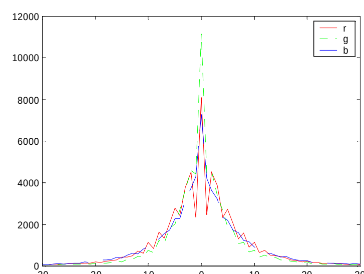
G	*i*	*i+1*	*i+2*	*i+3*	*i+4*
j	1.00	0.97	0.92	0.89	0.85
j+1	0.96	0.95	0.91	0.88	0.85
j+2	0.92	0.91	0.89	0.86	0.83
j+3	0.88	0.87	0.86	0.84	0.82
j+4	0.84	0.84	0.83	0.81	0.79

B	*i*	*i+1*	*i+2*	*i+3*	*i+4*
j	1.00	0.95	0.91	0.87	0.83
j+1	0.95	0.93	0.90	0.86	0.83
j+2	0.91	0.90	0.87	0.85	0.82
j+3	0.87	0.86	0.85	0.82	0.80
j+4	0.84	0.83	0.82	0.80	0.78

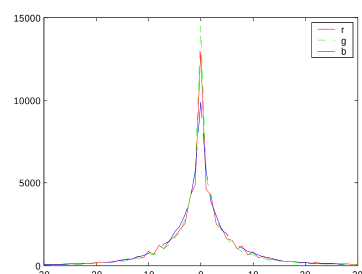
Normality of Differences of Adjacent Pixel Values



Spatial Difference Statistics: Tone

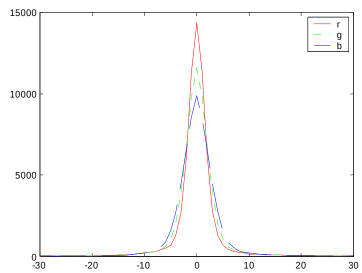


Rich tone in face/scarf

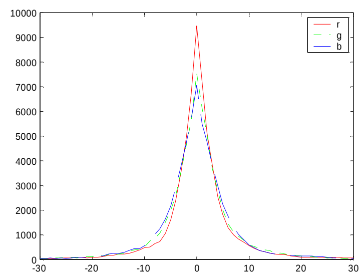


Tone lost in in shadows

Spatial Difference Statistics: Contrast

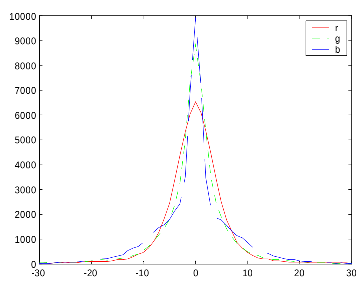


Low contrast

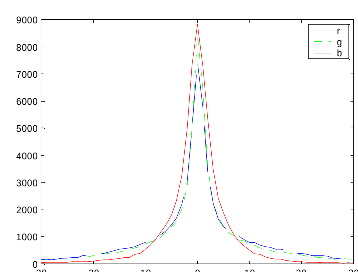


High contrast

Spatial Difference Statistics: Pattern

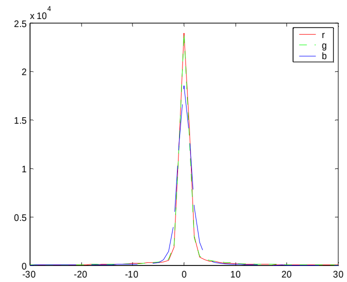


Clear separation between R and B

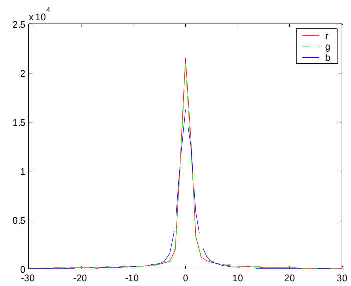


Large portion in shade

Spatial Difference Statistics: Pattern

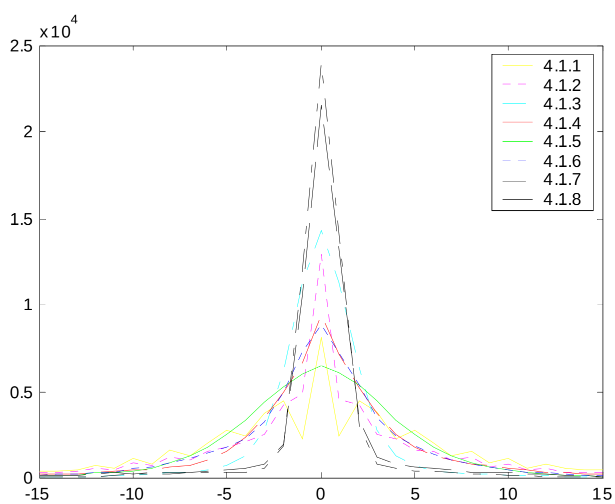


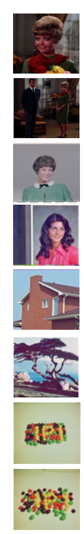
Plain tone



Irregular

Results Summary





- **Plot shows average of R, G, and B channels; Results are sorted**

Predicting the Pixel Values

	1-way	2-way	3-way	4-way
4.1.01	11.0130	8.7331	8.2357	8.0786
4.1.02	12.0754	8.2141	7.0998	7.0313
4.1.03	6.8144	6.2441	5.3144	5.2100
4.1.04	15.8660	8.4618	5.8766	5.8178
4.1.05	11.0146	8.6054	6.3802	6.2936
4.1.06	16.6531	13.5351	11.9507	11.7988
4.1.07	8.0694	5.8347	4.4617	4.3720
4.1.08	10.3856	7.7344	5.6498	5.4513

Residual Error of Multiple Regression Analysis

- **Use the neighboring pixels (white) to predict the current pixel (black)**
- **Run multiple regression analysis on the 8 benchmark images**
- **Prediction accuracy: 1-way < 2-way ~= 3-way ~= 4-way**
- **Hardware overhead is not linearly proportional to the # of ways**
  - **Large FIFO's are required to record RGB data of previous row (256*3)**
- **Hardware overhead: 1-way <<< 2-way ~= 3-way ~= 4-way. Therefore, <u>1-way pixel value prediction is chosen</u>**

Solution Technique Overview

- **Step 1) Spatial encoding is used to code the spatial differences between adjacent pixels in each color channel**
- **Step 2) Chromatic encoding is then applied to code the chromatic differences between the color channels resulting in encoded values with the least magnitudes**
- **Step 3) Ordered transition codes are exploited to encode the outputs of the previous encoding steps so as to minimize the intra-word switching activities in each DVI channel**
- **Steps 1 and 3 produce the energy optimal encoding of a single DVI channel**
- **Steps 1 thru 3 provide the energy optimal encoding of all three color channels in a DVI link**

Spatial Encoding

- For each channel, the spatial encoder
  - uses a one-way predictor to estimate the present pixel value (a trivial function, F, would predict that the current pixel value is the same as the previous pixel value; this is the one that we use)
  - outputs the difference between the actual and the predicted pixel value

Chromatic Correlation, Not

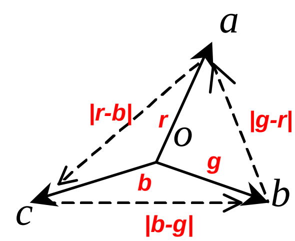
- In general, no correlation is found between the color channels
- We try to send values on the color channels that have the smallest magnitudes, because this will result in the minimum intra-word activity in each channel

CorrCoef	(R,G)	(G,B)	(B,R)
4.1.01	0.7712	0.9126	0.6819
4.1.02	0.8992	0.9478	0.8040
4.1.03	0.8579	0.9837	0.9098
4.1.04	0.6207	0.9274	0.6880
4.1.05	0.6378	0.9418	0.4823
4.1.06	0.0583	0.9736	0.0689
4.1.07	0.7016	0.8519	0.6478
4.1.08	0.6766	0.8481	0.6297

Correlation coefficients between R, G, and B values of each pixel

Chromatic Encoding and MST

- Let r, g, and b be the three values from the three color channels. Consider a weighted clique of four vertices {a, b, c, o} as shown
- Among these six edges, to represent the original values r, g, and b, exactly three edges are needed, and they must form a tree
- The optimal solution is a tree with the minimum weight, i.e., the minimum spanning tree (MST)



Chromatic Encoding Function

#	Ranking		Encoding					Decoding		
			α	β	γ	$z_0z_1z_2$	if sign()	r	g	b
1	0<r<g<b;	b<g<r<0	r	g-r	g-b	100	α=β	α	α+β	α+β–γ
2	0<r<b<g;	g<b<r<0	r	b-g	b-r	100	α=γ	α	α+γ–β	α+γ
3	0<g<b<r;	r<b<g<0	b-r	g	b-g	010	β=γ	β+γ–α	β	β+γ
4	0<g<r<b;	b<r<g<0	r-g	g	r-b	010	β=α	β+α	β	β+α–γ
5	0<b<r<g;	g<r<b<0	r-b	r-g	b	001	γ=α	γ+α	γ+α–β	γ
6	0<b<g<r;	r<g<b<0	g-r	g-b	b	001	γ=β	γ+β–α	γ+β	γ
7	b<0<g<r;	r<g<0<b	r-g	g	b	011	α=β	β+α	β	γ
8	g<0<b<r;	r<b<0<g	r-b	g	b	011	α=γ	γ+α	β	γ
9	r<0<b<g;	g<b<0<r	r	g-b	b	101	β=γ	α	β+γ	γ
10	b<0<r<g;	g<r<0<b	r	g-r	b	101	β=α	α	β+α	γ
11	g<0<r<b;	b<r<0<g	r	g	b-r	110	γ=α	α	β	γ+α
12	r<0<g<b;	b<g<0<r	r	g	b-g	110	γ=β	α	β	γ+β
13	overflow		r_0	g_0	b_0	000	-	α	β	γ
14	overflow		$TM(r_0)$	$TM(g_0)$	$TM(b_0)$	111	-	TM(α)	TM(β)	TM(γ)

Type 1: 0 < x < y < z ↔ **Type 3: z < y < x < 0** — *Z* = 100, 010, 001

Type 2: x < 0 < y < z ↔ **Type 4: z < y < 0 < x** — *Z* = 110, 011, 101

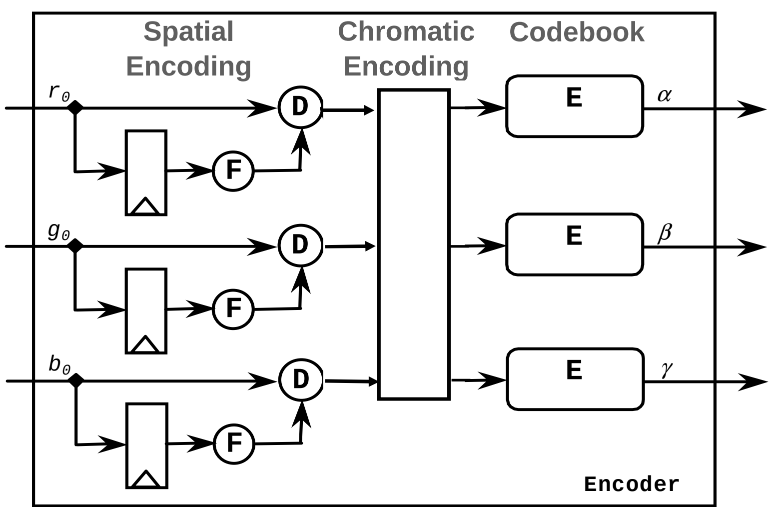
Ordered Transition Codes

- If the difference encoding is used, the total transition count can be written in terms of the signal differences ***x'***:

$$T = \sum_{x=-255}^{255} p(x')w(g(x'))$$

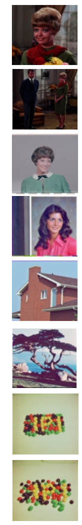
- ***p(x')***, which follows a Gaussian distribution, has the property that *p*(*x*)≤*p*(*y*) if |*x*|>|*y*|. The optimal code assignment will be any function ***g*** that satisfies *w*(*g*(*x*))≤*w*(*g*(*y*)) if |*x*|<|*y*|. Functions in this class are called ***ordered transition codes***
- The code assignment is generated as a two-column lookup table. In the first column, all of the source-words, ***x***, are sorted in increasing order of their magnitudes. In the second column, all of the code-words, ***e(x)***, (i.e., E[7..0] bits) are sorted in decreasing order of their intra-transition counts.

Complete Encoding Framework



Experimental Results: Images

	Spatial	Spatial+ Chromatic	Overflow
4.1.01	46.03%	54.36%	0
4.1.02	44.60%	54.70%	6
4.1.03	64.57%	72.83%	1
4.1.04	50.72%	60.83%	241
4.1.05	55.73%	62.96%	2
4.1.06	48.35%	57.61%	80
4.1.07	67.73%	73.20%	0
4.1.08	63.74%	69.98%	0



- **In both cases, we use OTC**
- **In case 2, E[8] in each color channel is replaced with the corresponding Z value**

Experimental Results: Video Clips

Clip name	*wg*	*tiger*	*final3*
Source type	clay animation	camcorder	anime
Frame number	331	634	1018
Frame size	304*224	320*240	160*128
Spatial	57.17%	38.89%	49.56%
Spatial+Chromatic	75.55%	64.47%	73.63%



Encoder Overhead

- Synopsys Design Analyzer
- VHDL
- TSMC 0.18 μm library
- Spatial Encoder: 12.52 mW
- Chromatic Encoder: 38.97 mW
- Codebook: 1.04 mW

Conclusions

- Proposed chromatic encoding for the DVI, a digital serial video interface
- Introduced tonal locality, the notion that the signal differences between adjacent pixels follow a Gaussian distribution
- Proved that spatial plus chromatic encoding reduces power consumption by minimizing the transition count on the DVI
- The proposed technique requires only three redundant bits for each 24-bit pixel, which are readily available
- Experimental results show up to a 75% transition reduction in the DVI