Document Image Analysis with Leptonica

Phototech EDU, 4 April 07

Dan Bloomberg
bloomberg@ieee.org

Introduction

- Image analysis in a course in photographic technology?
- Image analysis in the last century.
- Hofstadter’s 100 milliseconds and image processing.
- Trade-off between speed and accuracy.
- Two examples of scaling
  - Linear interpolation on color
  - Rank order cascade of 2x reductions on binary
- Why *document* image analysis?
  - Easier than natural scenes
  - Useful: conversion from paper to digital
  - Interesting: input is not well-defined

Roadmap

Outline of talk

- Goals
  - Page information extraction
  - Restoration and/or appearance improvement
  - Compression
- Approach
  - Nonlinear/Shape and Texture/Use the image
- Primary tools
  - Image morphology
  - Affine transforms
  - Counting and components
  - Seedfill
  - Leptonica library
- Example applications
  - Page image segmentation
  - Background cleaning of bad photocopy
  - Skew, keystoning and baselines
  - Unsupervised shape classification
  - Color segmentation/quantization

● Introduction

Roadmap

Goals

● Page information extraction

d/or● appearance improvement

● Compression

Approach

Primary tools

Example Applications

Goals

Page information extraction

- Global information
  - Skew and text orientation
  - Non-affine warping (e.g., projective)
- Components on the page
  - Text, image, rules, ...
  - What are they?
  - Where are they located?
  - What is the hierarchical arrangement?
  - What are the equivalence classes?
- Photometry
  - What is the background color?
  - Are there color images?

Restoration and/or appearance improvement

- Geometrical
  - Image deskew
  - Global dewarping
- Color mapping
  - Set background to uniform color
  - Compensate for lighting variations
  - Map text to increase contrast; preserve antialiasing
  - Map images for larger dynamic range
  - Detect and remove color moire
- Other
  - Remove noise from binary scans
  - Remove bleedthrough
  - Scale to gray for display
  - Interpolated upscaling for print
  - Quantization for compression

Compression

- Artifacts
    - JPEG 8x8 block noise near text
    - Color moire: alias on halftones and gravure
    - Binary thresholding
        - Increases contrast: bad for images
        - Removes antialias: bad for text at low resolution
- Avoidance techniques
    - Uniform background
    - Quantization of text
    - Capture at higher resolution
    - Demosaic to gray if no color
    - Mixed raster output

Approach

Approach

- Nonlinear: decisions made on each pixel
  - Linear operations don’t make decisions
  - Implicit labels assigned to pixels
  - Bottom-up aggregation
- Extraction of shape and texture
  - Shape at one scale is texture at another
  - Work at appropriate scale
  - Use morphology to seive
  - Use morphology and rank reductions to modify texture
  - Use seedfill for robust segmentation and labelling
- Image as primary representation
  - All the information is there – don’t lose it
  - Use image processing to do (nearly) everything
  - Complex, difficult and limiting to use other representations
  - Simple, easy and general to visualize imaging methods

Primary tools

Image morphology (1)

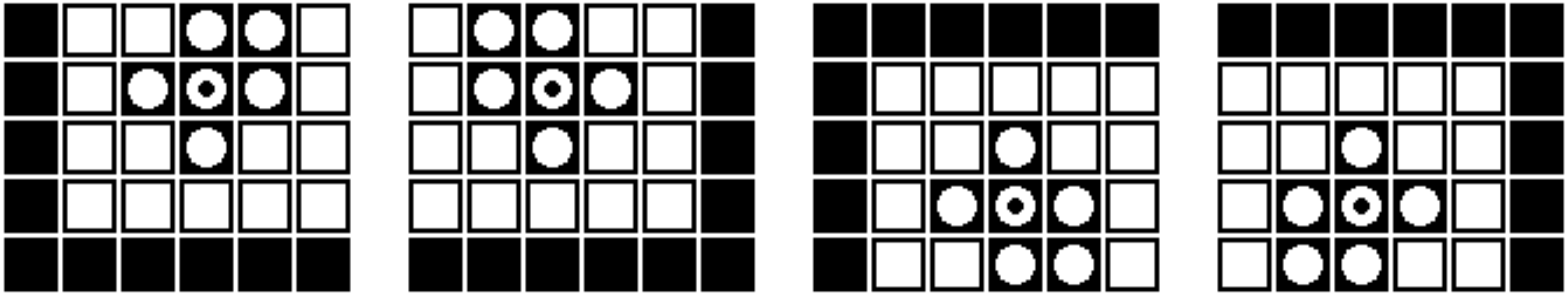
- References
  - *www.leptonica.org/binary-morphology.html*
  - *www.leptonica.org/papers/morphdefs.pdf*
- What is it?
  - Method for extracting shape and texture
  - Image processing operations: dilation and erosion
  - Analogy with convolution
  - Nonlinear: special case of rank order filters
    - Dilation is MAX, Erosion is MIN
  - Kernel is Sel ("structuring element")
    - Hits, misses, don’t-cares, origin
  - Opening and closing are composite operations
    - idempotent; independent of origin
  - Dualities
  - Hit-miss operation is general pattern match

Image morphology (2)

- Historical

  Invented in France in the 60s
  Very slow adoption in the US

- Example of hit-miss Sels



- These are used to identify character ascenders and descenders

Image morphology (3)

- Implementation through rasterop
  - Always use packed images and full word operations
  - Conceptual: test Sel at each point on src
  - Actual: let Sel direct full image rasterops
    - Erosion: copy first; then AND
    - Dilation: OR each hit
  - Efficiency for brick Sels
    - Separable in x and y
    - Composable as sequence at different scales
- Implementation through dwa (dest word accumulation)
  - Reference: *www.leptonica.org/papers/binmorph.pdf*
  - Auto-gen’d code
  - Unrolled destination word loop
  - typically 3-4x faster than rasterop
- Both can be invoked for brick Sels with an interpreter.

Affine transforms (1)

- Translation: rasterop
- Shear: rasterop
- Rotation
  - Reference: *www.leptonica.org/rotation.html*
  - By rasterop: 2 shear and 3 shear
  - By area mapping (linear interpolation)
- Scaling
  - Reference: *www.leptonica.org/scaling.html*
  - Useful for many things
    - Rendering: interpolation up; antialias down
    - Combining with depth change for rendering
    - Choosing scale at which to work
    - Combining morphology with subsampling: texture filtering

Affine transforms (2)

- Scaling types
  - Binary to gray (downscale)
    - example: display high res binary on screen as grayscale
  - Gray to binary (upscale)
    - example: convert to high res binary for print, display
  - Gray to gray
  - Binary to binary
- Binary to binary: rank order 2x cascade
  - Generalization of morphology + subsampling
  - Useful for texture filtering
  - Fast word parallel operation
  - Rank = 1 (1 or more are fg) solidifies fg
  - Rank = 4 (all 4 are fg) erodes fg

Counting and components

- Fg pixels in 1 bpp images
  - Test for *any* fg pixels
  - Sum pixels on raster scanlines
    - Use for determining skew
- Connected components in 1 bpp images
  - Use for labeling components
  - Use for adaptive thresholding; e.g., word segmentation
- Histograms in 8 bpp images
  - Attach tentative labels (text, image)
  - Generate 1 bpp masks

Seedfill

- Use to label connected components
  - Remove components sequentially
  - Optionally save component bitmap
- Requires seed and mask images
  - Fill into seed; clip to mask
- Slow, parallel, morphological method
  - Iterate with 3x3 brick Sel for 8-c.c. fill
  - Number of iterations depends on component size
- Fast, sequential, raster/anti-raster fill
  - Use for all full-image seedfill
  - Typically requires several pairs of traverses
  - Number of iterations is independent of component size
- Grayscale version exists
  - Fast, sequential, raster/anti-raster fill
  - Use for analyzing peaks

Leptonica library (1)

- Lightweight (efficient) C library
  - Mostly low-level imaging functions
  - Written in 2001 - 2003; maintained to present
  - Works with both endians
  - About 20 structs, 1000 functions
  - Open source
  - Most parts have been extensively tested
  - Tailored for document image analysis
  - The image is the primary object
  - Available at:
    - *www.leptonica.org*
    - *code.google.com/p/leptonica*
    - *debian packages: libleptonica, etc.*

Leptonica library (2)

- Basic infrastructure
  - rasterop (depth independent)
  - affine transforms
    - – scaling, translation, rotation, shear
    - – on all depths; often with or without colormaps
  - binary morphology (two different implementations)
  - grayscale morphology and convolution
  - connected components and sequential seedfill
  - transforms combining changes in scale and pixel depth
  - pixelwise masking, blending, enhancement, arith ops, etc.
  - I/O for jpeg, png, tiff, pnm, bmp; O for PostScript
  - lots more

Leptonica library (3)

- Various "applications"
  - octcube-based color quantization (incl. dithering)
  - skew determination of doc images
  - segmentation of page images with mixed text/images
  - jbig2 unsupervised classifier
  - border representations of bitmaps; raster conversion
  - PostScript wrapping of images (levels 1,2)
  - playing around (e.g., least-cost paths in images)

● Introduction

Roadmap

Goals

Approach

Primary tools

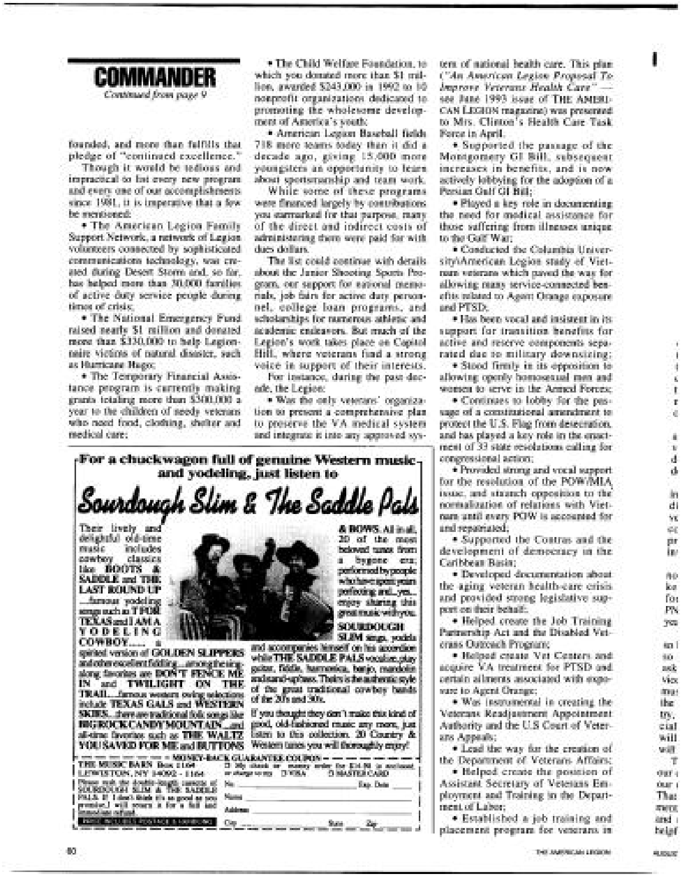
Example Applications

● Page segmentation (1)
● Page segmentation (2)
● Page segmentation (3)
● Page segmentation (4)
● Page segmentation (5)
● Page segmentation (6)
● Page segmentation (7)
● Page segmentation (8)
● Page segmentation (9)
● Page segmentation (10)
● Page segmentation (11)
● Page segmentation (12)
eaning of bad photocopy (1)
eaning of bad photocopy (2)
eaning of bad photocopy (3)
● Deskew by differential line sums (1)
● Deskew by differential line sums (2)
● Deskew by differential line sums (3)
● Keystoning and baselines (1)
● Keystoning and baselines (2)
● Keystoning and baselines (3)
● Keystoning and baselines (4)

Example Applications

Page segmentation (1)

- First identify halftone image regions
- Then identify text lines
- Then aggregate into text blocks



COMMANDER

Continued from page 9

founded, and more than fulfills that pledge of "continued excellence."

Though it would be tedious and impractical to list every new program and every one of our accomplishments since 1981, it is imperative that a few be mentioned:

• The American Legion Family Support Network, a network of Legion volunteers connected by sophisticated communications technology, was created during Desert Storm and, so far, has helped more than 30,000 families of active duty service people during times of crisis;

• The National Emergency Fund raised nearly $1 million and donated more than $330,000 to help Legionnaire victims of natural disaster, such as Hurricane Hugo;

• The Temporary Financial Assistance program is currently making grants totaling more than $300,000 a year to the children of needy veterans who need food, clothing, shelter and medical care;

• The Child Welfare Foundation, to which you donated more than $1 million, awarded $243,000 in 1992 to 10 nonprofit organizations dedicated to promoting the wholesome development of America's youth;

• American Legion Baseball fields 718 more teams today than it did a decade ago, giving 15,000 more youngsters an opportunity to learn about sportsmanship and team work.

While some of these programs were financed largely by contributions you earmarked for that purpose, many of the direct and indirect costs of administering them were paid for with dues dollars.

The list could continue with details about the Junior Shooting Sports Program, our support for national memorials, job fairs for active duty personnel, college loan programs, and scholarships for numerous athletic and academic endeavors. But much of the Legion's work takes place on Capitol Hill, where veterans find a strong voice in support of their interests.

For instance, during the past decade, the Legion:

• Was the only veterans' organization to present a comprehensive plan to preserve the VA medical system and integrate it into any approved system of national health care. This plan (*"An American Legion Proposal To Improve Veterans Health Care"* — see June 1993 issue of THE AMERICAN LEGION magazine) was presented to Mrs. Clinton's Health Care Task Force in April.

• Supported the passage of the Montgomery GI Bill, subsequent increases in benefits, and is now actively lobbying for the adoption of a Persian Gulf GI Bill;

• Played a key role in documenting the need for medical assistance for those suffering from illnesses unique to the Gulf War;

• Conducted the Columbia University/American Legion study of Vietnam veterans which paved the way for allowing many service-connected benefits related to Agent Orange exposure and PTSD;

• Has been vocal and insistent in its support for transition benefits for active and reserve components separated due to military downsizing;

• Stood firmly in its opposition to allowing openly homosexual men and women to serve in the Armed Forces;

• Continues to lobby for the passage of a constitutional amendment to protect the U.S. Flag from desecration, and has played a key role in the enactment of 33 state resolutions calling for congressional action;

• Provided strong and vocal support for the resolution of the POW/MIA issue, and staunch opposition to the normalization of relations with Vietnam until every POW is accounted for and repatriated;

• Supported the Contras and the development of democracy in the Caribbean Basin;

• Developed documentation about the aging veteran health-care crisis and provided strong legislative support on their behalf;

• Helped create the Job Training Partnership Act and the Disabled Veterans Outreach Program;

• Helped create Vet Centers and acquire VA treatment for PTSD and certain ailments associated with exposure to Agent Orange;

• Was instrumental in creating the Veterans Readjustment Appointment Authority and the U.S Court of Veterans Appeals;

• Lead the way for the creation of the Department of Veterans Affairs;

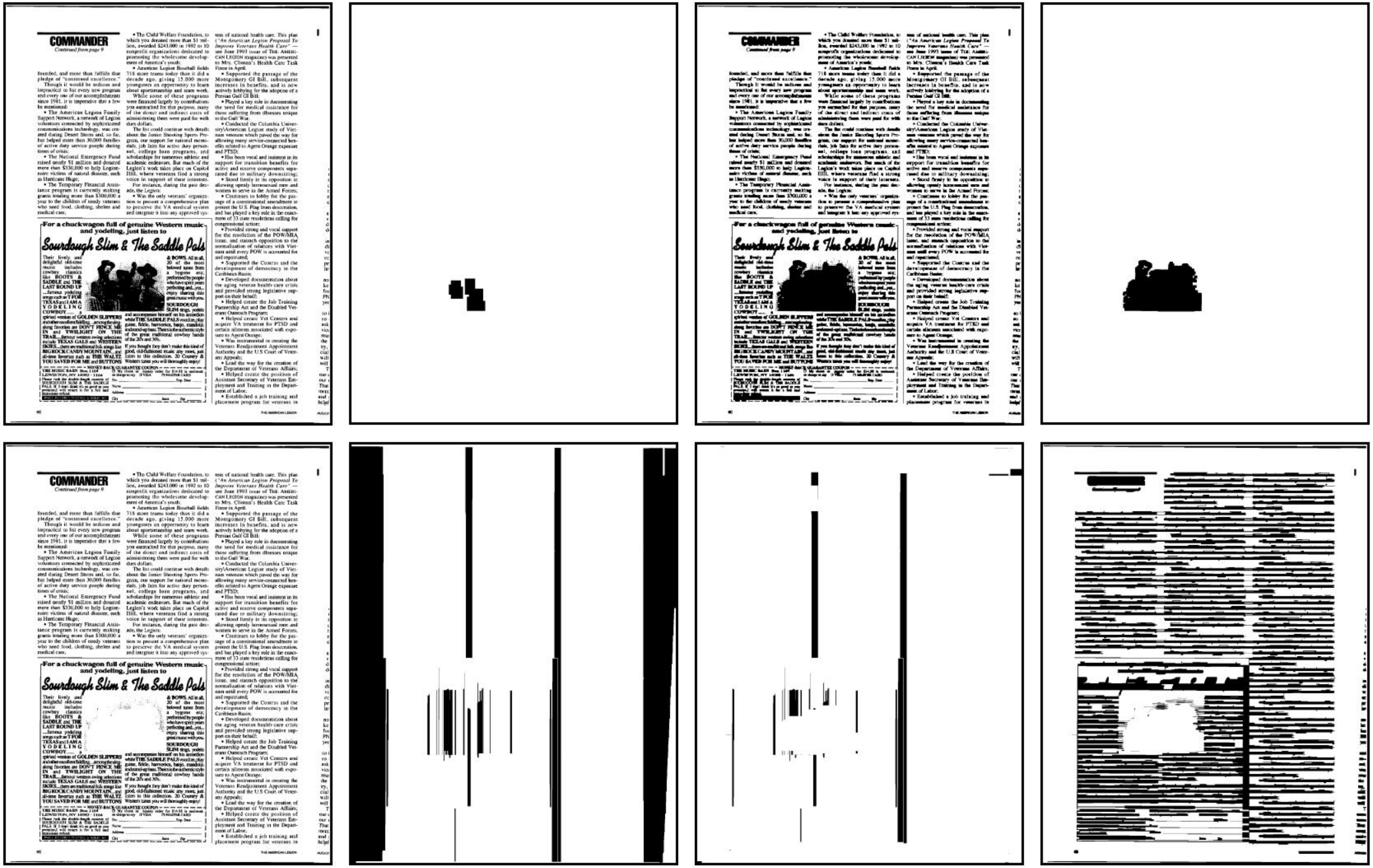
• Helped create the position of Assistant Secretary of Veterans Employment and Training in the Department of Labor;

• Established a job training and placement program for veterans in

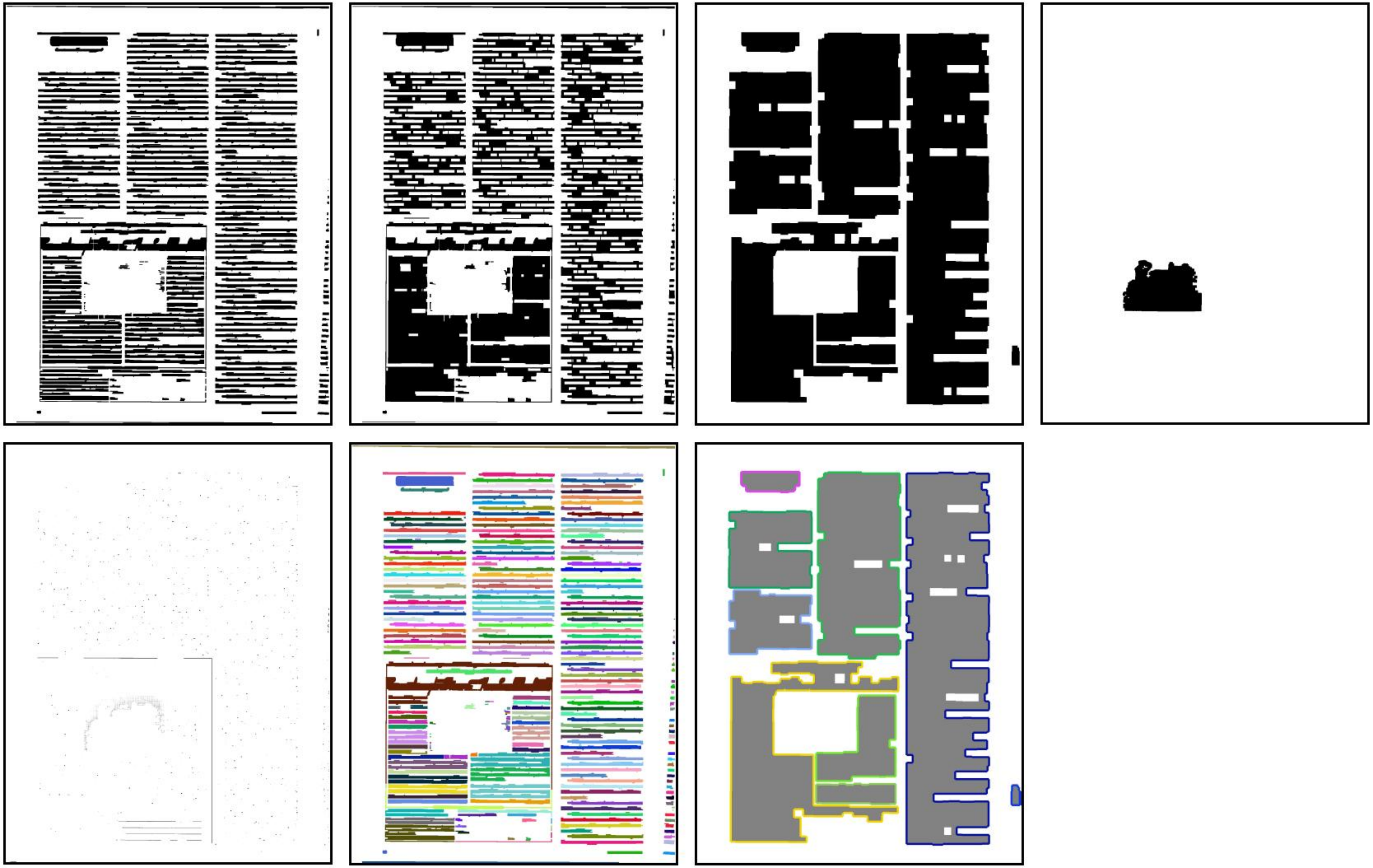
80

THE AMERICAN LEGION

Page segmentation (2)

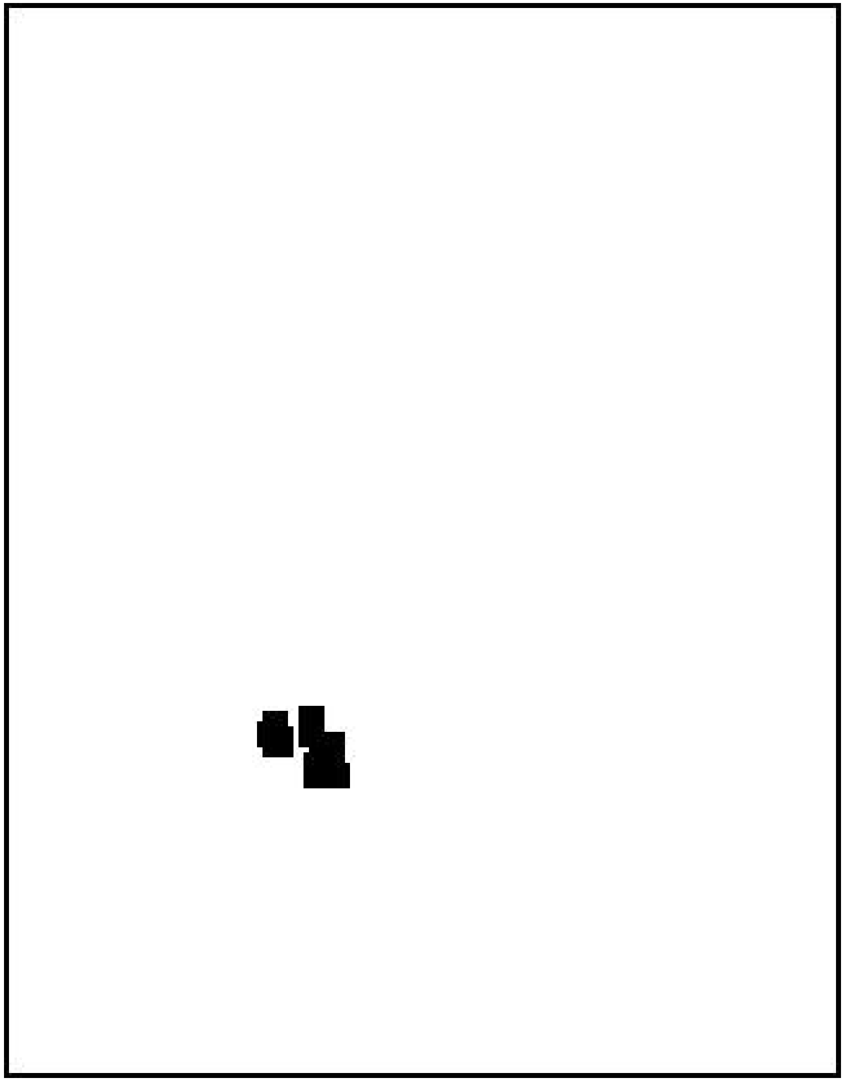


Page segmentation (3)



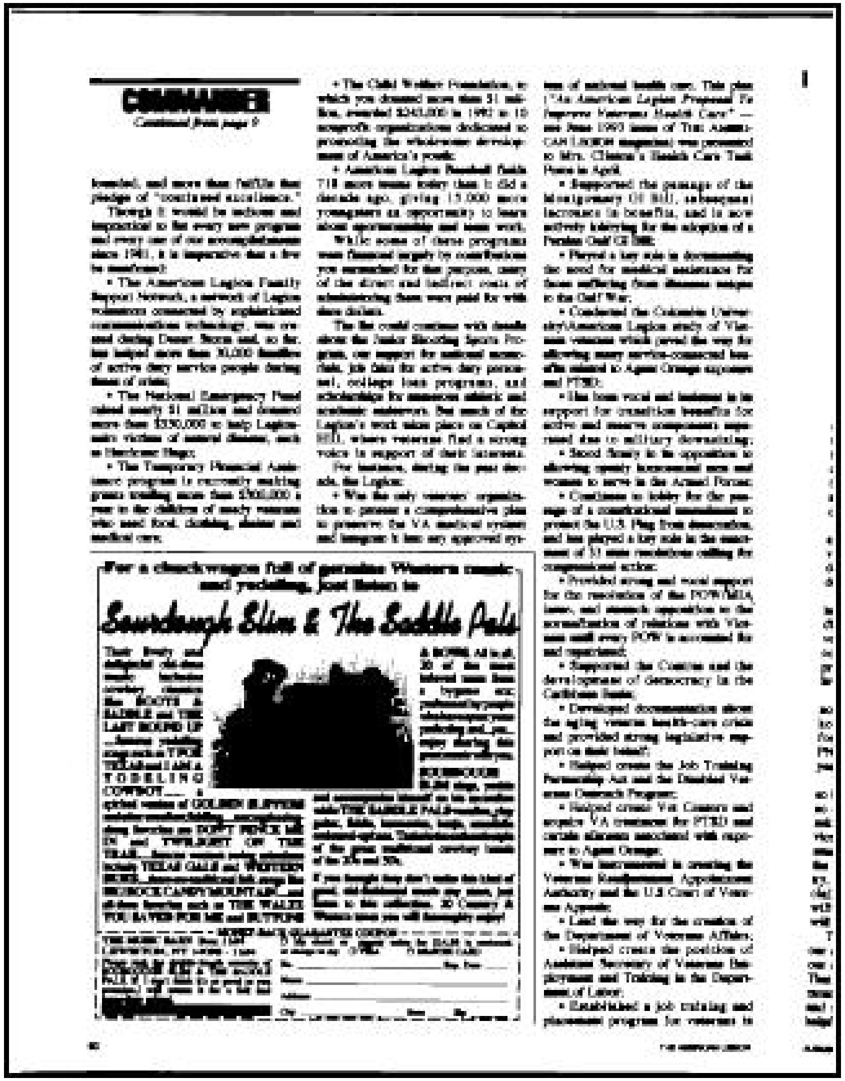
Page segmentation (4)

- `pixt1 = pixReduceRankBinaryCascade (pixs, 4, 4, 3, 0);`
- `pixt2 = pixOpenBrick (NULL, pixt1, 5, 5);`
- `pixhs = pixExpandBinary (pixt2, 8);`



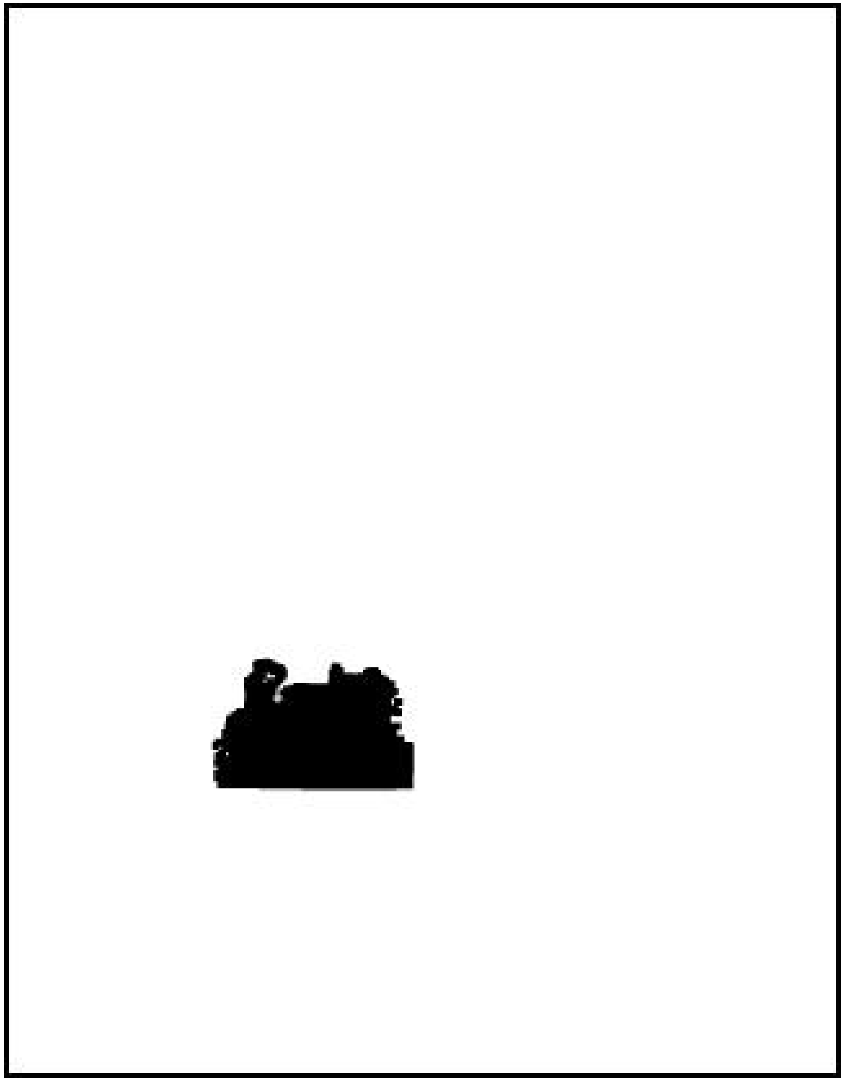
Page segmentation (5)

- `pixm = pixCloseSafeBrick (NULL, pixs, 4, 4);`



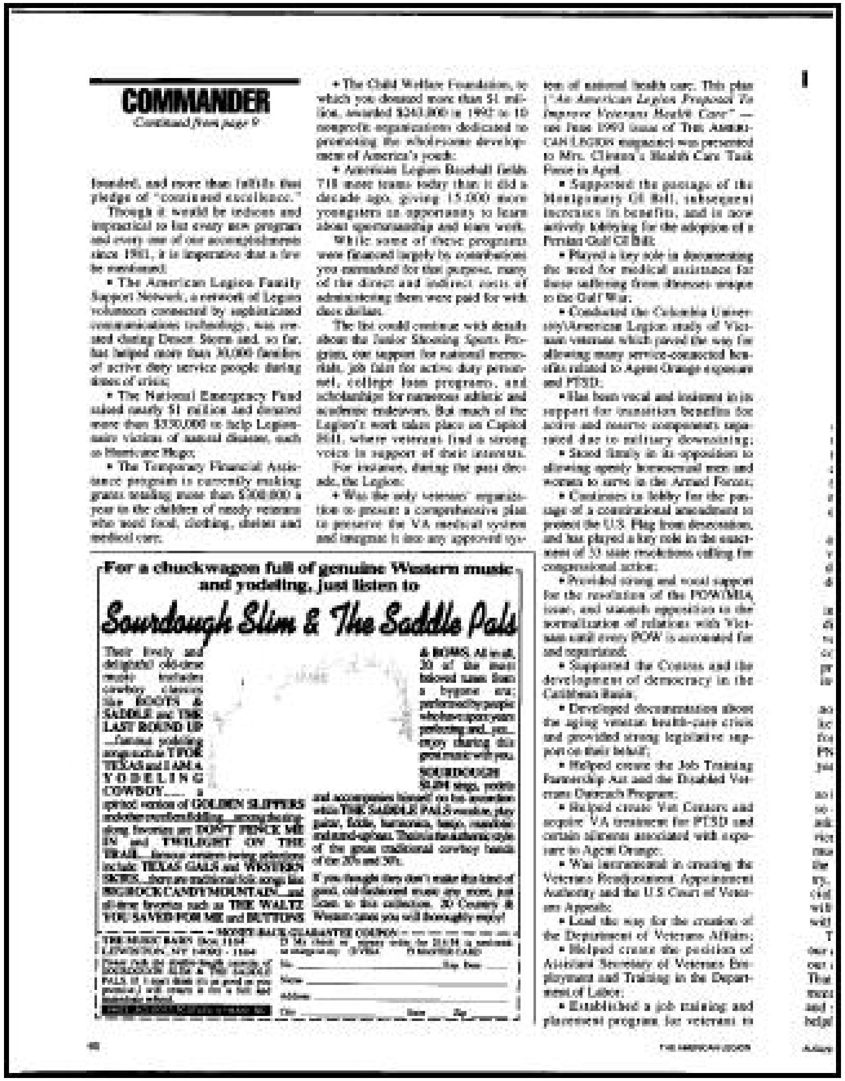
Page segmentation (6)

- pixhm = pixSeedfillBinary (NULL, pixhs, pixm, 4); // open to remove small lines, etc.
- pixOpenBrick (pixhm, pixhm, 10, 10);



Page segmentation (7)

- `pixtext = pixSubtract (NULL, pixs, pixhm);`



COMMANDER

Continued from page 9

founded, and more than fulfills that pledge of "continued excellence."

Though it would be tedious and impractical to list every new program and every one of our accomplishments since 1981, it is imperative that a few be mentioned:

• The American Legion Family Support Network, a network of Legion volunteers connected by sophisticated communications technology, was created during Desert Storm and, so far, has helped more than 30,000 families of active duty service people during times of crisis;

• The National Emergency Fund raised nearly $1 million and donated more than $330,000 to help Legionnaire victims of natural disasters, such as Hurricane Hugo;

• The Temporary Financial Assistance program is currently making grants totaling more than $300,000 a year to the children of needy veterans who need food, clothing, shelter and medical care.

• The Child Welfare Foundation, to which you donated more than $1 million, awarded $243,000 in 1992 to 10 nonprofit organizations dedicated to promoting the wholesome development of America's youth;

• American Legion Baseball fields 718 more teams today than it did a decade ago, giving 15,000 more youngsters an opportunity to learn about sportsmanship and team work.

While some of these programs were financed largely by contributions you earmarked for that purpose, many of the direct and indirect costs of administering them were paid for with dues dollars.

The list could continue with details about the Junior Shooting Sports Program, our support for national memorials, job fairs for active duty personnel, college loan programs, and scholarships for numerous athletic and academic endeavors. But much of the Legion's work takes place on Capitol Hill, where veterans find a strong voice in support of their interests.

For instance, during the past decade, the Legion:

• Was the only veterans' organization to present a comprehensive plan to preserve the VA medical system and integrate it into any approved system of national health care. This plan (*"An American Legion Proposal To Improve Veterans Health Care"* — see June 1993 issue of THE AMERICAN LEGION magazine) was presented to Mrs. Clinton's Health Care Task Force in April.

• Supported the passage of the Montgomery GI Bill, subsequent increases in benefits, and is now actively lobbying for the adoption of a Persian Gulf GI Bill;

• Played a key role in documenting the need for medical assistance for those suffering from illnesses unique to the Gulf War;

• Conducted the Columbia University/American Legion study of Vietnam veterans which paved the way for allowing many service-connected benefits related to Agent Orange exposure and PTSD;

• Has been vocal and insistent in its support for transition benefits for active and reserve components separated due to military downsizing;

• Stood firmly in its opposition to allowing openly homosexual men and women to serve in the Armed Forces;

• Continues to lobby for the passage of a constitutional amendment to protect the U.S. Flag from desecration, and has played a key role in the enactment of 33 state resolutions calling for congressional action;

• Provided strong and vocal support for the resolution of the POW/MIA issue, and staunch opposition to the normalization of relations with Vietnam until every POW is accounted for and repatriated;

• Supported the Contras and the development of democracy in the Caribbean Basin;

• Developed documentation about the aging veteran health-care crisis and provided strong legislative support on their behalf;

• Helped create the Job Training Partnership Act and the Disabled Veterans Outreach Program;

• Helped create Vet Centers and acquire VA treatment for PTSD and certain ailments associated with exposure to Agent Orange;

• Was instrumental in creating the Veterans Readjustment Appointment Authority and the U.S. Court of Veterans Appeals;

• Lead the way for the creation of the Department of Veterans Affairs;

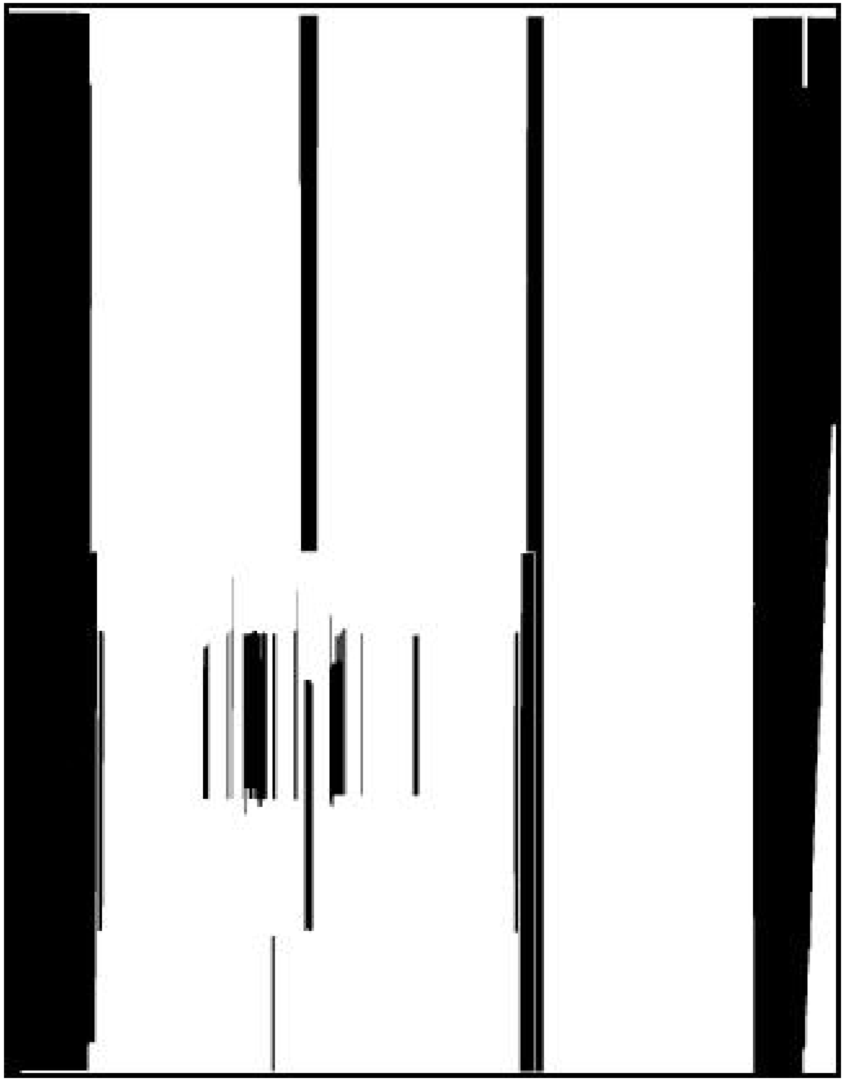
• Helped create the position of Assistant Secretary of Veterans Employment and Training in the Department of Labor;

• Established a job training and placement program for veterans to

48 THE AMERICAN LEGION

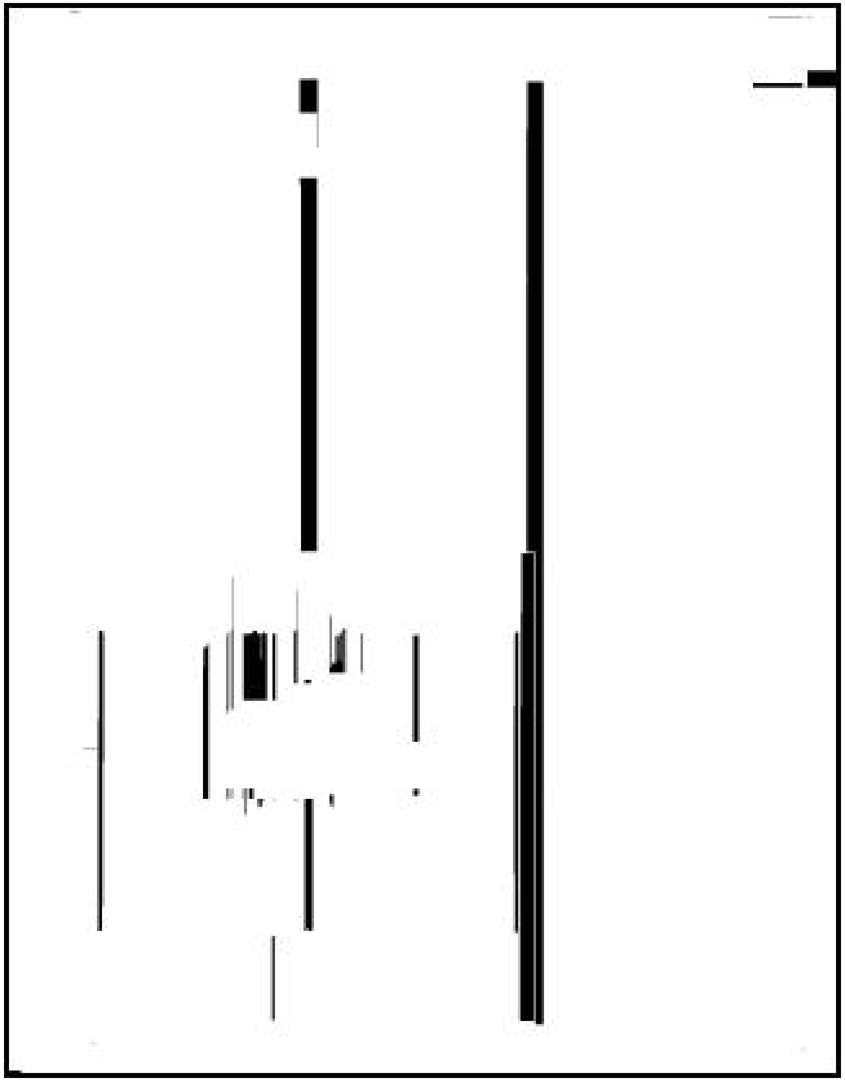
Page segmentation (8)

- pixinv = pixInvert (NULL, pixs);
- pixvws = pixMorphCompSequence (pixinv, "o5.1 + 01.200", 0);



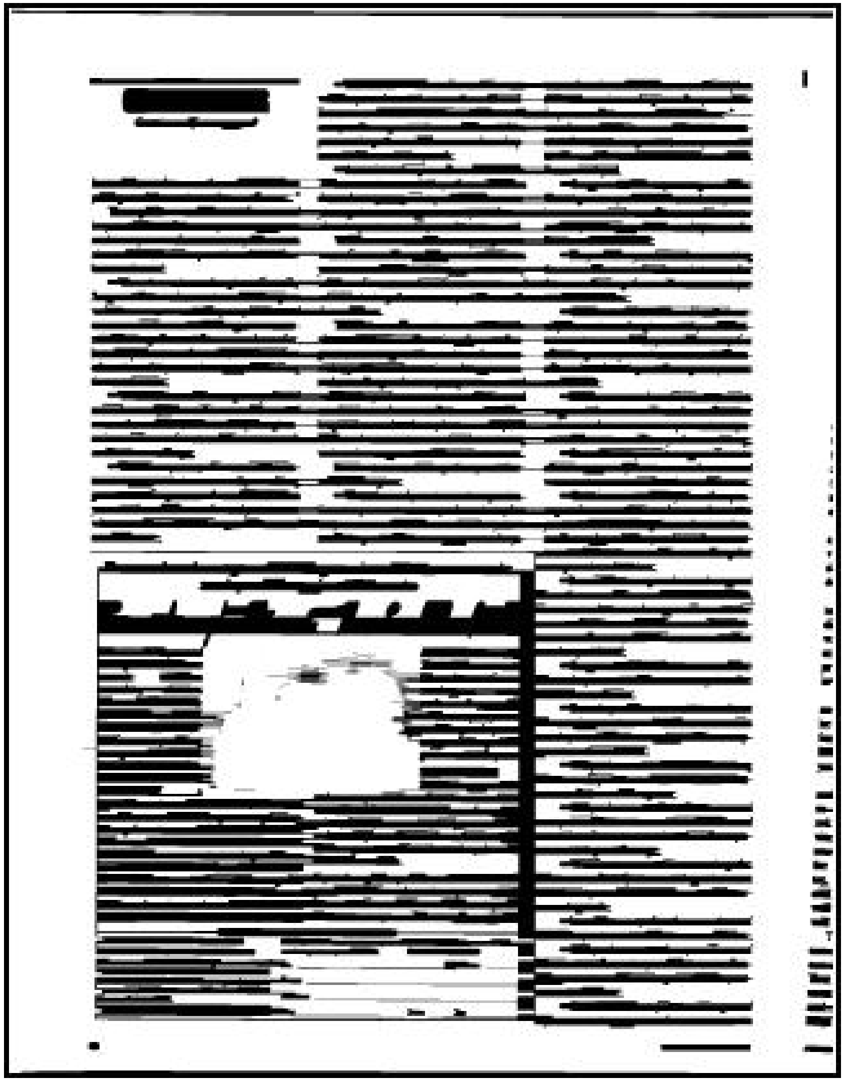
Page segmentation (9)

- `pixt1 = pixMorphCompSequence(pixinv, "o80.60", 0);`
- `pixSubtract (pixvws, pixvws, pixt1);`
- `pixDestroy (&pixt1);`



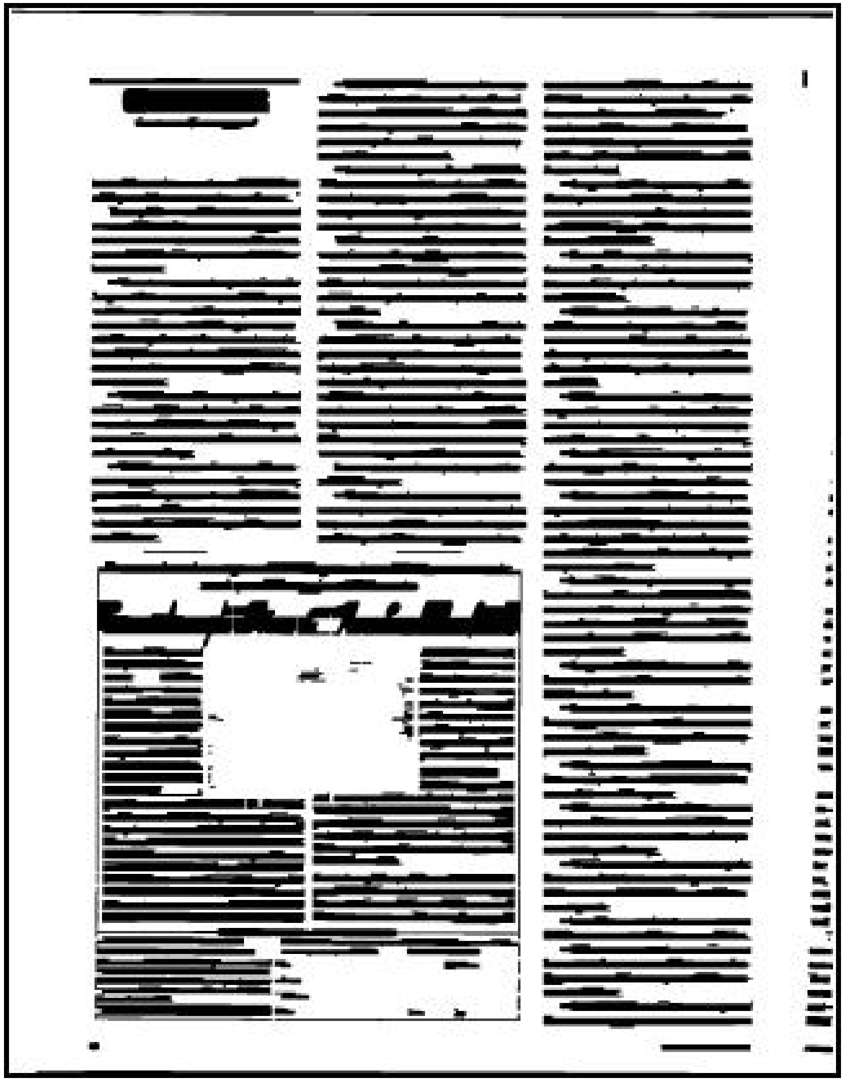
Page segmentation (10)

- `pixt1 = pixCloseSafeBrick (NULL, pixs, 30, 1);`



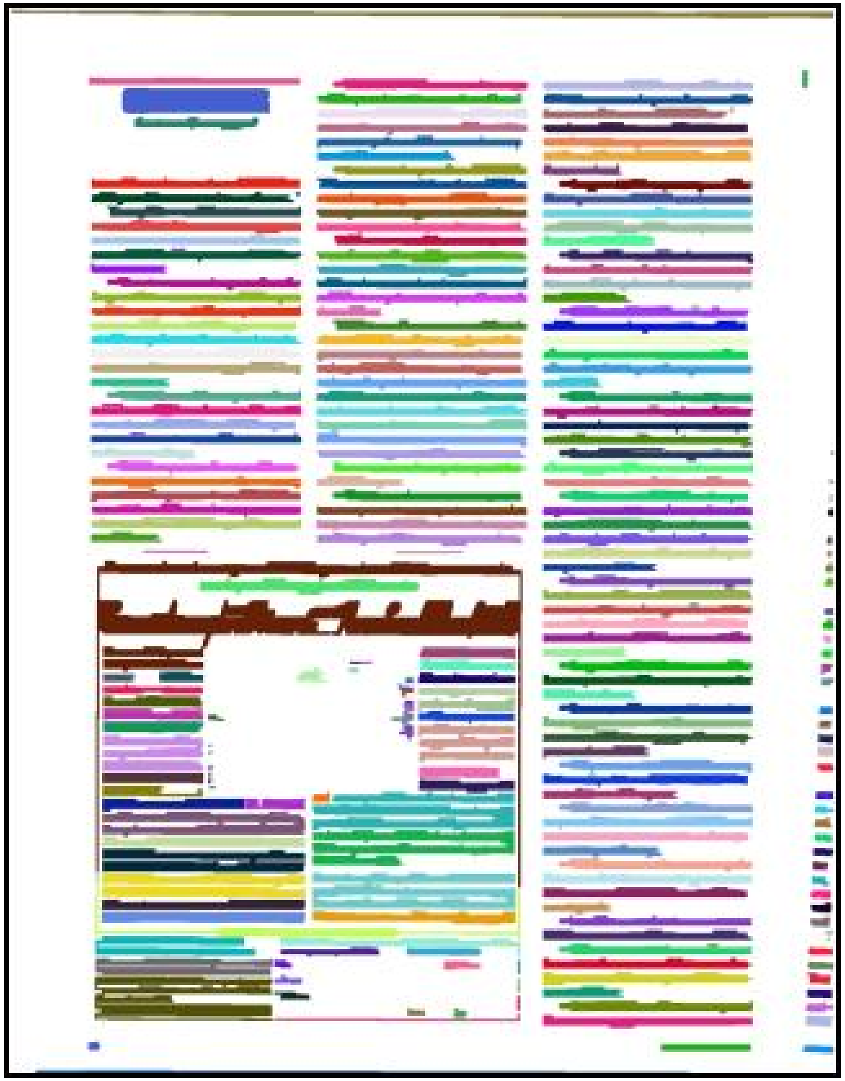
Page segmentation (11)

- `pixlines = pixSubtract (NULL, pixt1, pixvws);`
- `pixOpenBrick (pixlines, pixlines, 3, 3);`



Page segmentation (12)

- Boxa *boxa = pixConnComp (pixlines, &pixa, 8);
- pixGetDimensions (pixlines, &w, &h, NULL);
- pixc = pixaDisplayRandomCmap(pixa, w, h);
- pixcmapResetColor (pixGetColormap(pixc), 0, 255, 255, 255);



Background cleaning of bad photocopy (1)

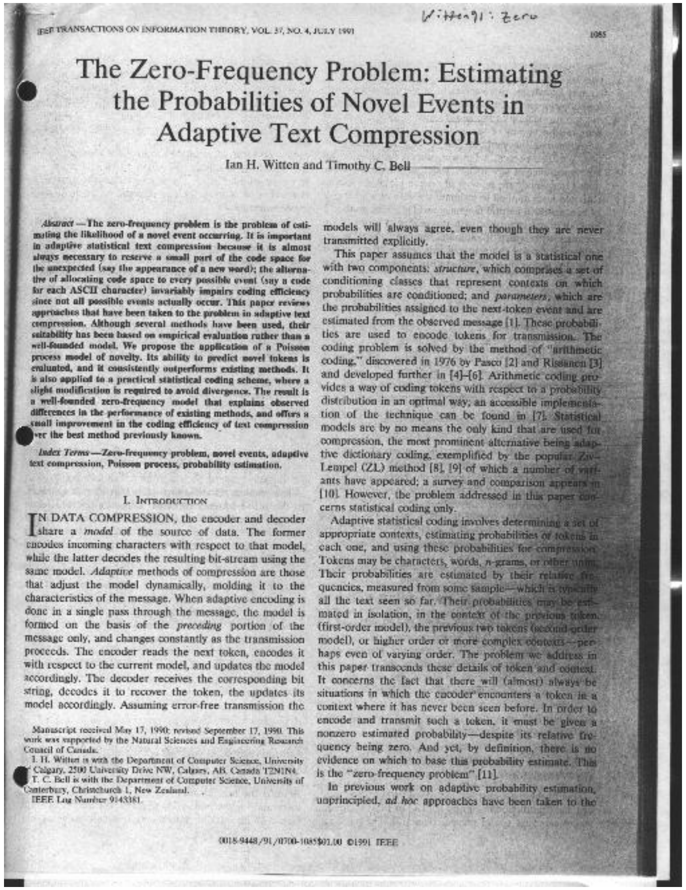
- Adaptive background normalization
    - More flexible than background thresholding
    - Two methods to get background values
        - Morphological closing to remove foreground
        - Tiling, bg estimation, filling, smoothing
    - Map pixel values locally
        - Background goes to fixed global value
- Threshold to get binary output if desired
- Simple method for computing background

```
pixs = pixRead ("contrast-orig-60.jpg");
pixt1 = pixCloseGray (pixs, 11, 11);
```

or: pixt1 = pixScaleGrayMinMax (pixs, 11, 11, L_CHOOSE_MAX);

```
pixt2 = pixBlockconv(pixt1, 15, 15);
```

Background cleaning of bad photocopy (2)



Witten91: Zero

The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression

Ian H. Witten and Timothy C. Bell

Abstract—The zero-frequency problem is the problem of estimating the likelihood of a novel event occurring. It is important in adaptive statistical text compression because it is almost always necessary to reserve a small part of the code space for the unexpected (say the appearance of a new word); the alternative of allocating code space to every possible event (say a code for each ASCII character) invariably impairs coding efficiency since not all possible events actually occur. This paper reviews approaches that have been taken to the problem in adaptive text compression. Although several methods have been used, their suitability has been based on empirical evaluation rather than a well-founded model. We propose the application of a Poisson process model of novelty. Its ability to predict novel tokens is evaluated, and it consistently outperforms existing methods. It is also applied to a practical statistical coding scheme, where a slight modification is required to avoid divergence. The result is a well-founded zero-frequency model that explains observed differences in the performance of existing methods, and offers a small improvement in the coding efficiency of text compression over the best method previously known.

Index Terms—Zero-frequency problem, novel events, adaptive text compression, Poisson process, probability estimation.

I. Introduction

IN DATA COMPRESSION, the encoder and decoder share a *model* of the source of data. The former encodes incoming characters with respect to that model, while the latter decodes the resulting bit-stream using the same model. *Adaptive* methods of compression are those that adjust the model dynamically, molding it to the characteristics of the message. When adaptive encoding is done in a single pass through the message, the model is formed on the basis of the *preceding* portion of the message only, and changes constantly as the transmission proceeds. The encoder reads the next token, encodes it with respect to the current model, and updates the model accordingly. The decoder receives the corresponding bit string, decodes it to recover the token, the updates its model accordingly. Assuming error-free transmission the models will always agree, even though they are never transmitted explicitly.

This paper assumes that the model is a statistical one with two components: *structure*, which comprises a set of conditioning classes that represent contexts on which probabilities are conditioned; and *parameters*, which are the probabilities assigned to the next-token event and are estimated from the observed message [1]. These probabilities are used to encode tokens for transmission. The coding problem is solved by the method of "arithmetic coding," discovered in 1976 by Pasco [2] and Rissanen [3] and developed further in [4]–[6]. Arithmetic coding provides a way of coding tokens with respect to a probability distribution in an optimal way; an accessible implementation of the technique can be found in [7]. Statistical models are by no means the only kind that are used for compression, the most prominent alternative being adaptive dictionary coding, exemplified by the popular Ziv–Lempel (ZL) method [8], [9] of which a number of variants have appeared; a survey and comparison appears in [10]. However, the problem addressed in this paper concerns statistical coding only.

Adaptive statistical coding involves determining a set of appropriate contexts, estimating probabilities of tokens in each one, and using these probabilities for compression. Tokens may be characters, words, *n*-grams, or other units. Their probabilities are estimated by their relative frequencies, measured from some sample—which is typically all the text seen so far. Their probabilities may be estimated in isolation, in the context of the previous token (first-order model), the previous two tokens (second-order model), or higher order or more complex contexts—perhaps even of varying order. The problem we address in this paper transcends these details of token and context. It concerns the fact that there will (almost) always be situations in which the encoder encounters a token in a context where it has never been seen before. In order to encode and transmit such a token, it must be given a nonzero estimated probability—despite its relative frequency being zero. And yet, by definition, there is no evidence on which to base this probability estimate. This is the "zero-frequency problem" [11].

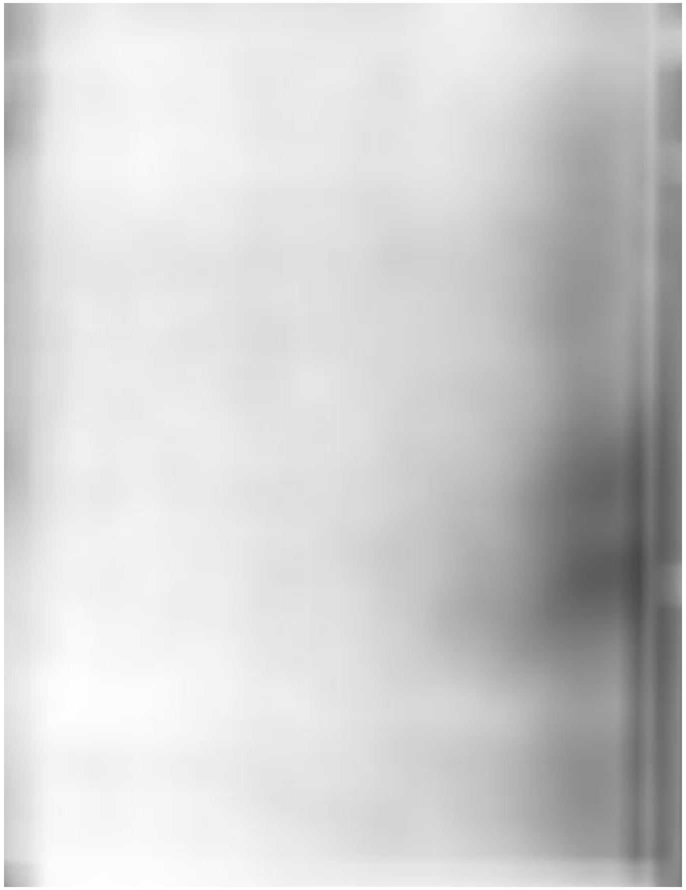
In previous work on adaptive probability estimation, unprincipled, *ad hoc* approaches have been taken to the

Manuscript received May 17, 1990; revised September 17, 1990. This work was supported by the Natural Sciences and Engineering Research Council of Canada.

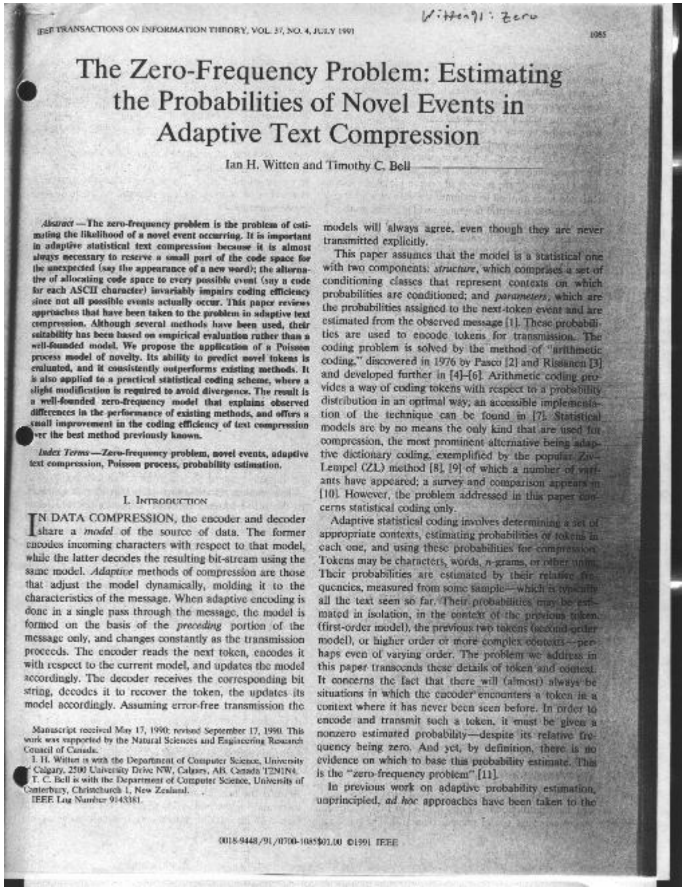
I. H. Witten is with the Department of Computer Science, University of Calgary, 2500 University Drive NW, Calgary, AB, Canada T2N1N4.
T. C. Bell is with the Department of Computer Science, University of Canterbury, Christchurch 1, New Zealand.

IEEE Log Number 9143181.

0018-9448/91/0700-1085$01.00 ©1991 IEEE



Background cleaning of bad photocopy (3)



IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 37, NO. 4, JULY 1991 1085

Witten91 : Zero

The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression

Ian H. Witten and Timothy C. Bell

Abstract—The zero-frequency problem is the problem of estimating the likelihood of a novel event occurring. It is important in adaptive statistical text compression because it is almost always necessary to reserve a small part of the code space for the unexpected (say the appearance of a new word); the alternative of allocating code space to every possible event (say a code for each ASCII character) invariably impairs coding efficiency since not all possible events actually occur. This paper reviews approaches that have been taken to the problem in adaptive text compression. Although several methods have been used, their suitability has been based on empirical evaluation rather than a well-founded model. We propose the application of a Poisson process model of novelty. Its ability to predict novel tokens is evaluated, and it consistently outperforms existing methods. It is also applied to a practical statistical coding scheme, where a slight modification is required to avoid divergence. The result is a well-founded zero-frequency model that explains observed differences in the performance of existing methods, and offers a small improvement in the coding efficiency of text compression over the best method previously known.

Index Terms—Zero-frequency problem, novel events, adaptive text compression, Poisson process, probability estimation.

I. Introduction

IN DATA COMPRESSION, the encoder and decoder share a *model* of the source of data. The former encodes incoming characters with respect to that model, while the latter decodes the resulting bit-stream using the same model. *Adaptive* methods of compression are those that adjust the model dynamically, molding it to the characteristics of the message. When adaptive encoding is done in a single pass through the message, the model is formed on the basis of the *preceding* portion of the message only, and changes constantly as the transmission proceeds. The encoder reads the next token, encodes it with respect to the current model, and updates the model accordingly. The decoder receives the corresponding bit string, decodes it to recover the token, the updates its model accordingly. Assuming error-free transmission the models will always agree, even though they are never transmitted explicitly.

This paper assumes that the model is a statistical one with two components: *structure*, which comprises a set of conditioning classes that represent contexts on which probabilities are conditioned; and *parameters*, which are the probabilities assigned to the next-token event and are estimated from the observed message [1]. These probabilities are used to encode tokens for transmission. The coding problem is solved by the method of "arithmetic coding," discovered in 1976 by Pasco [2] and Rissanen [3] and developed further in [4]–[6]. Arithmetic coding provides a way of coding tokens with respect to a probability distribution in an optimal way; an accessible implementation of the technique can be found in [7]. Statistical models are by no means the only kind that are used for compression, the most prominent alternative being adaptive dictionary coding, exemplified by the popular Ziv–Lempel (ZL) method [8], [9] of which a number of variants have appeared; a survey and comparison appears in [10]. However, the problem addressed in this paper concerns statistical coding only.

Adaptive statistical coding involves determining a set of appropriate contexts, estimating probabilities of tokens in each one, and using these probabilities for compression. Tokens may be characters, words, *n*-grams, or other units. Their probabilities are estimated by their relative frequencies, measured from some sample—which is typically all the text seen so far. Their probabilities may be estimated in isolation, in the context of the previous token (first-order model), the previous two tokens (second-order model), or higher order or more complex contexts—perhaps even of varying order. The problem we address in this paper transcends these details of token and context. It concerns the fact that there will (almost) always be situations in which the encoder encounters a token in a context where it has never been seen before. In order to encode and transmit such a token, it must be given a nonzero estimated probability—despite its relative frequency being zero. And yet, by definition, there is no evidence on which to base this probability estimate. This is the "zero-frequency problem" [11].

In previous work on adaptive probability estimation, unprincipled, *ad hoc* approaches have been taken to the

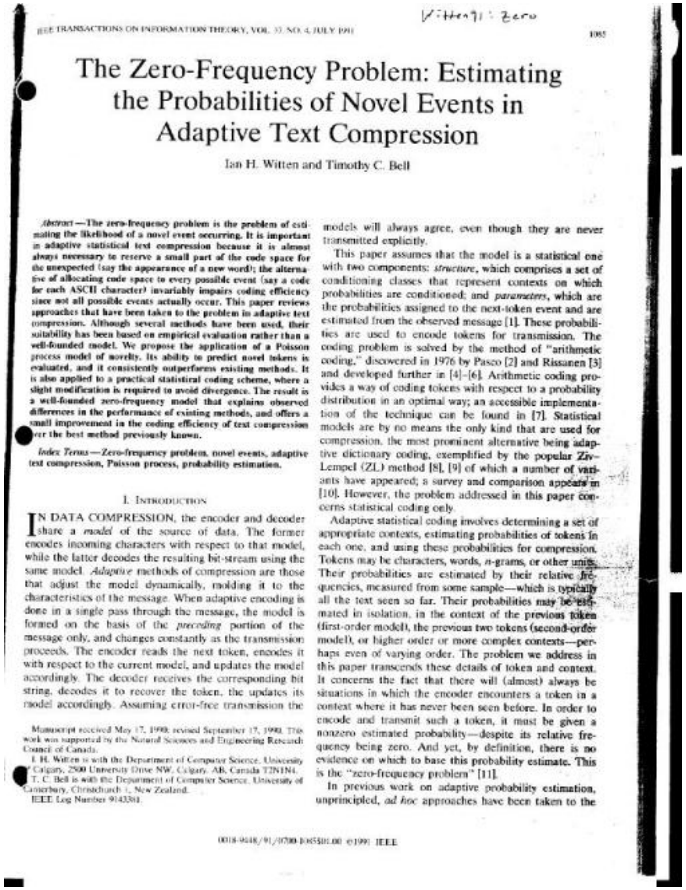
Manuscript received May 17, 1990; revised September 17, 1990. This work was supported by the Natural Sciences and Engineering Research Council of Canada.

I. H. Witten is with the Department of Computer Science, University of Calgary, 2500 University Drive NW, Calgary, AB, Canada T2N1N4.

T. C. Bell is with the Department of Computer Science, University of Canterbury, Christchurch 1, New Zealand.

IEEE Log Number 9143381.

0018-9448/91/0700-1085$01.00 ©1991 IEEE



IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 37, NO. 4, JULY 1991 1085

Witten91 : Zero

The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression

Ian H. Witten and Timothy C. Bell

Abstract—The zero-frequency problem is the problem of estimating the likelihood of a novel event occurring. It is important in adaptive statistical text compression because it is almost always necessary to reserve a small part of the code space for the unexpected (say the appearance of a new word); the alternative of allocating code space to every possible event (say a code for each ASCII character) invariably impairs coding efficiency since not all possible events actually occur. This paper reviews approaches that have been taken to the problem in adaptive text compression. Although several methods have been used, their suitability has been based on empirical evaluation rather than a well-founded model. We propose the application of a Poisson process model of novelty. Its ability to predict novel tokens is evaluated, and it consistently outperforms existing methods. It is also applied to a practical statistical coding scheme, where a slight modification is required to avoid divergence. The result is a well-founded zero-frequency model that explains observed differences in the performance of existing methods, and offers a small improvement in the coding efficiency of text compression over the best method previously known.

Index Terms—Zero-frequency problem, novel events, adaptive text compression, Poisson process, probability estimation.

I. Introduction

IN DATA COMPRESSION, the encoder and decoder share a *model* of the source of data. The former encodes incoming characters with respect to that model, while the latter decodes the resulting bit-stream using the same model. *Adaptive* methods of compression are those that adjust the model dynamically, molding it to the characteristics of the message. When adaptive encoding is done in a single pass through the message, the model is formed on the basis of the *preceding* portion of the message only, and changes constantly as the transmission proceeds. The encoder reads the next token, encodes it with respect to the current model, and updates the model accordingly. The decoder receives the corresponding bit string, decodes it to recover the token, the updates its model accordingly. Assuming error-free transmission the models will always agree, even though they are never transmitted explicitly.

This paper assumes that the model is a statistical one with two components: *structure*, which comprises a set of conditioning classes that represent contexts on which probabilities are conditioned; and *parameters*, which are the probabilities assigned to the next-token event and are estimated from the observed message [1]. These probabilities are used to encode tokens for transmission. The coding problem is solved by the method of "arithmetic coding," discovered in 1976 by Pasco [2] and Rissanen [3] and developed further in [4]–[6]. Arithmetic coding provides a way of coding tokens with respect to a probability distribution in an optimal way; an accessible implementation of the technique can be found in [7]. Statistical models are by no means the only kind that are used for compression, the most prominent alternative being adaptive dictionary coding, exemplified by the popular Ziv–Lempel (ZL) method [8], [9] of which a number of variants have appeared; a survey and comparison appears in [10]. However, the problem addressed in this paper concerns statistical coding only.

Adaptive statistical coding involves determining a set of appropriate contexts, estimating probabilities of tokens in each one, and using these probabilities for compression. Tokens may be characters, words, *n*-grams, or other units. Their probabilities are estimated by their relative frequencies, measured from some sample—which is typically all the text seen so far. Their probabilities may be estimated in isolation, in the context of the previous token (first-order model), the previous two tokens (second-order model), or higher order or more complex contexts—perhaps even of varying order. The problem we address in this paper transcends these details of token and context. It concerns the fact that there will (almost) always be situations in which the encoder encounters a token in a context where it has never been seen before. In order to encode and transmit such a token, it must be given a nonzero estimated probability—despite its relative frequency being zero. And yet, by definition, there is no evidence on which to base this probability estimate. This is the "zero-frequency problem" [11].

In previous work on adaptive probability estimation, unprincipled, *ad hoc* approaches have been taken to the

Manuscript received May 17, 1990; revised September 17, 1990. This work was supported by the Natural Sciences and Engineering Research Council of Canada.

I. H. Witten is with the Department of Computer Science, University of Calgary, 2500 University Drive NW, Calgary, AB, Canada T2N1N4.

T. C. Bell is with the Department of Computer Science, University of Canterbury, Christchurch 1, New Zealand.

IEEE Log Number 9143381.

0018-9448/91/0700-1085$01.00 ©1991 IEEE

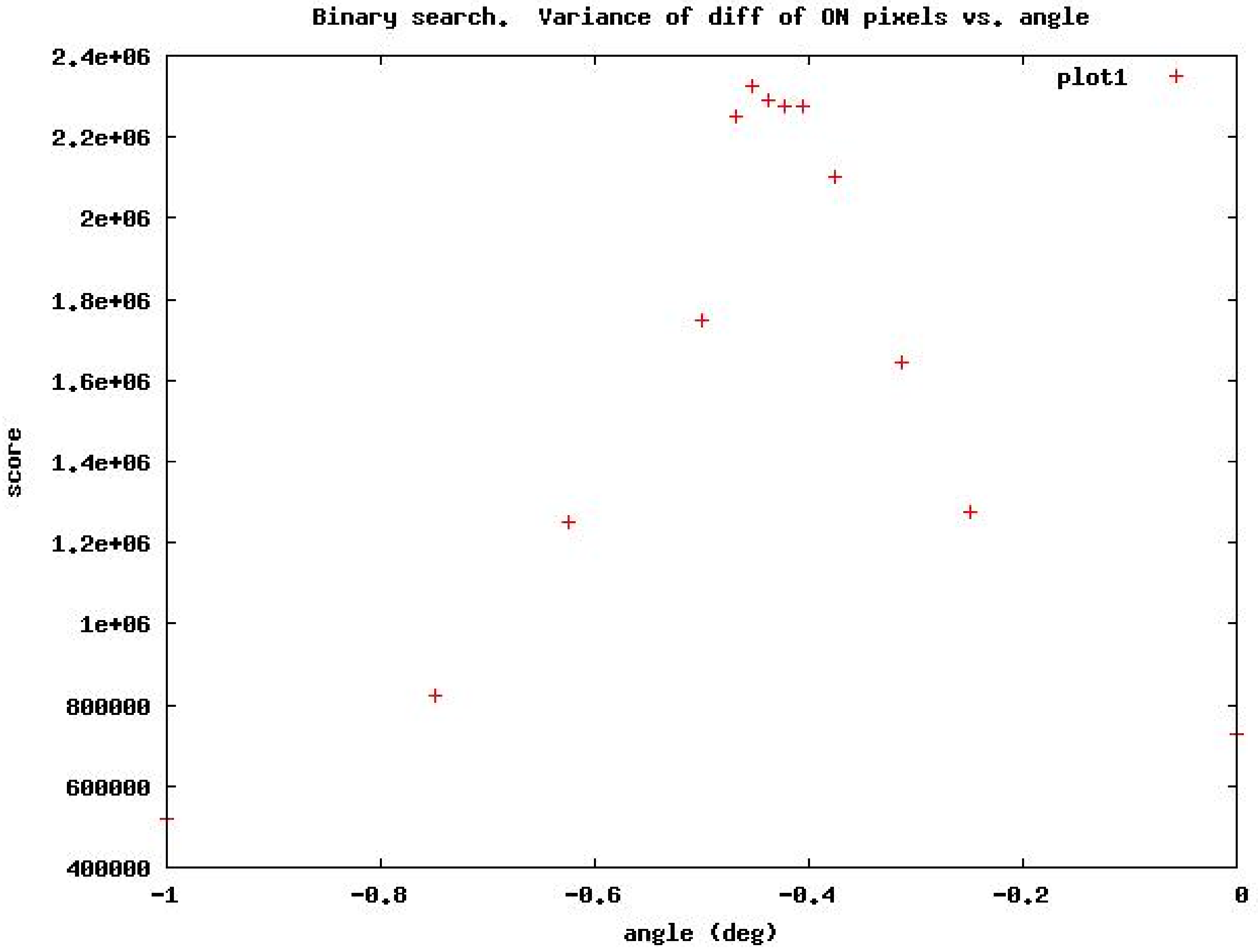
Deskew by differential line sums (1)

- References

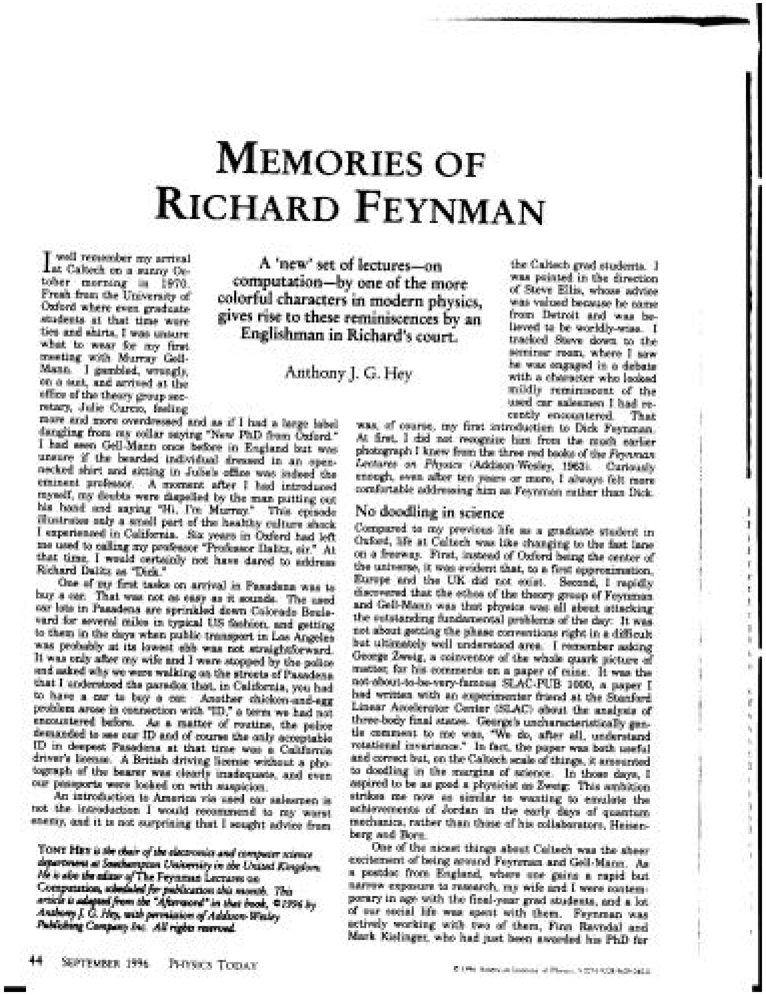
  www.leptonica.org/skew-measurement.html (general background)

  www.leptonica.org/papers/docskew.pdf (technical description)
- Most robust method (Postl, 1988)
- Use vertical shear to mimic rotation
- Maximize variance of difference of line sums on adjacent lines
- Use coarse linear search followed by binary search
- Typically compute at 100 - 150 ppi resolution
- Accuracy approximately 1 vertical pixel: 1/w in radians
- This is about 0.05 degree
- People do not notice angles less than about 0.2 degree

Deskew by differential line sums (2)



Deskew by differential line sums (3)



MEMORIES OF RICHARD FEYNMAN

A 'new' set of lectures—on computation—by one of the more colorful characters in modern physics, gives rise to these reminiscences by an Englishman in Richard's court.

Anthony J. G. Hey

I well remember my arrival at Caltech on a sunny October morning in 1970. Fresh from the University of Oxford where even graduate students at that time wore ties and shirts, I was unsure what to wear for my first meeting with Murray Gell-Mann. I gambled, wrongly, on a suit, and arrived at the office of the theory group secretary, Julie Curcio, feeling more and more overdressed and as if I had a large label dangling from my collar saying "New PhD from Oxford." I had seen Gell-Mann once before in England but was unsure if the bearded individual dressed in an open-necked shirt and sitting in Julie's office was indeed the eminent professor. A moment after I had introduced myself, my doubts were dispelled by the man putting out his hand and saying "Hi, I'm Murray." This episode illustrates only a small part of the healthy culture shock I experienced in California. Six years in Oxford had left me used to calling my professor "Professor Dalitz, sir." At that time, I would certainly not have dared to address Richard Dalitz as "Dick."

One of my first tasks on arrival in Pasadena was to buy a car. That was not as easy as it sounds. The used car lots in Pasadena are sprinkled down Colorado Boulevard for several miles in typical US fashion, and getting to them in the days when public transport in Los Angeles was probably at its lowest ebb was not straightforward. It was only after my wife and I were stopped by the police and asked why we were walking on the streets of Pasadena that I understood the paradox that, in California, you had to have a car to buy a car. Another chicken-and-egg problem arose in connection with "ID," a term we had not encountered before. As a matter of routine, the police demanded to see our ID and of course the only acceptable ID in deepest Pasadena at that time was a California driver's license. A British driving license without a photograph of the bearer was clearly inadequate, and even our passports were looked on with suspicion.

An introduction to America via used car salesmen is not the introduction I would recommend to my worst enemy, and it is not surprising that I sought advice from the Caltech grad students. I was pointed in the direction of Steve Ellis, whose advice was valued because he came from Detroit and was believed to be worldly-wise. I tracked Steve down to the seminar room, where I saw he was engaged in a debate with a character who looked mildly reminiscent of the used car salesmen I had recently encountered. That was, of course, my first introduction to Dick Feynman. At first, I did not recognize him from the much earlier photograph I knew from the three red books of the *Feynman Lectures on Physics* (Addison-Wesley, 1963). Curiously enough, even after ten years or more, I always felt more comfortable addressing him as Feynman rather than Dick.

No doodling in science

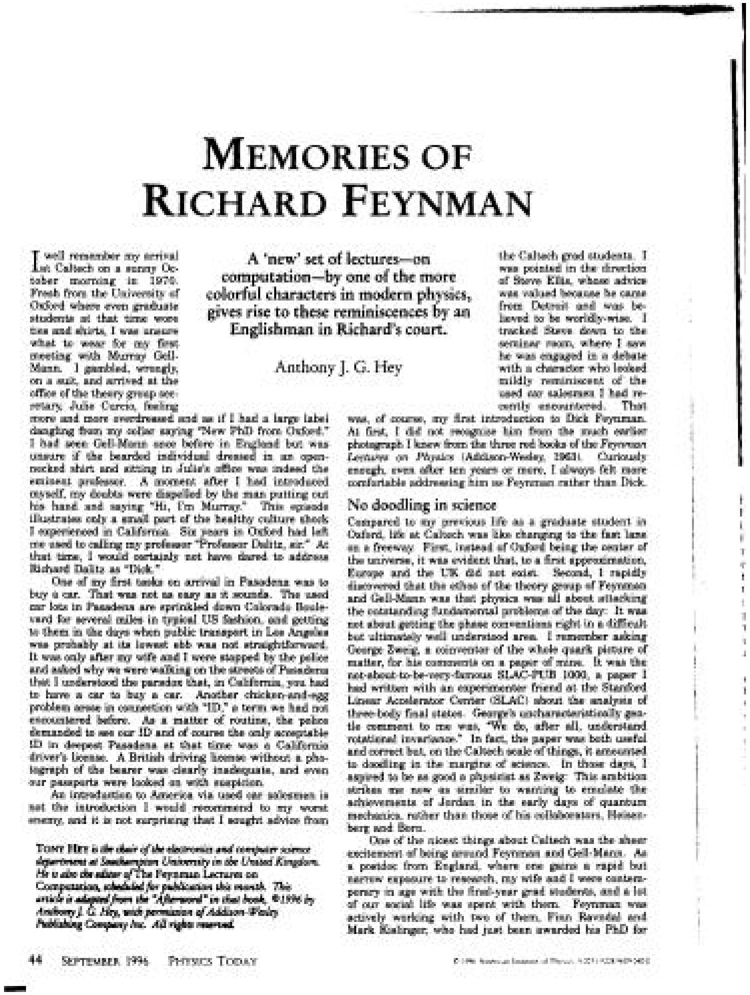
Compared to my previous life as a graduate student in Oxford, life at Caltech was like changing to the fast lane on a freeway. First, instead of Oxford being the center of the universe, it was evident that, to a first approximation, Europe and the UK did not exist. Second, I rapidly discovered that the ethos of the theory group of Feynman and Gell-Mann was that physics was all about attacking the outstanding fundamental problems of the day: It was not about getting the phase conventions right in a difficult but ultimately well understood area. I remember asking George Zweig, a coinventor of the whole quark picture of matter, for his comments on a paper of mine. It was the not-about-to-be-very-famous SLAC-PUB 1000, a paper I had written with an experimenter friend at the Stanford Linear Accelerator Center (SLAC) about the analysis of three-body final states. George's uncharacteristically gentle comment to me was, "We do, after all, understand rotational invariance." In fact, the paper was both useful and correct but, on the Caltech scale of things, it amounted to doodling in the margins of science. In those days, I aspired to be as good a physicist as Zweig: This ambition strikes me now as similar to wanting to emulate the achievements of Jordan in the early days of quantum mechanics, rather than those of his collaborators, Heisenberg and Born.

One of the nicest things about Caltech was the sheer excitement of being around Feynman and Gell-Mann. As a postdoc from England, where one gains a rapid but narrow exposure to research, my wife and I were contemporary in age with the final-year grad students, and a lot of our social life was spent with them. Feynman was actively working with two of them, Finn Ravndal and Mark Kislinger, who had just been awarded his PhD for

TONY HEY is the chair of the electronics and computer science department at Southampton University in the United Kingdom. He is also the editor of The Feynman Lectures on Computation, *scheduled for publication this month. This article is adapted from the "Afterword" in that book, ©1996 by Anthony J. G. Hey, with permission of Addison-Wesley Publishing Company Inc. All rights reserved.*

44 SEPTEMBER 1996 PHYSICS TODAY

© 1996 American Institute of Physics, [illegible]



MEMORIES OF RICHARD FEYNMAN

A 'new' set of lectures—on computation—by one of the more colorful characters in modern physics, gives rise to these reminiscences by an Englishman in Richard's court.

Anthony J. G. Hey

I well remember my arrival at Caltech on a sunny October morning in 1970. Fresh from the University of Oxford where even graduate students at that time wore ties and shirts, I was unsure what to wear for my first meeting with Murray Gell-Mann. I gambled, wrongly, on a suit, and arrived at the office of the theory group secretary, Julie Curcio, feeling more and more overdressed and as if I had a large label dangling from my collar saying "New PhD from Oxford." I had seen Gell-Mann once before in England but was unsure if the bearded individual dressed in an open-necked shirt and sitting in Julie's office was indeed the eminent professor. A moment after I had introduced myself, my doubts were dispelled by the man putting out his hand and saying "Hi, I'm Murray." This episode illustrates only a small part of the healthy culture shock I experienced in California. Six years in Oxford had left me used to calling my professor "Professor Dalitz, sir." At that time, I would certainly not have dared to address Richard Dalitz as "Dick."

One of my first tasks on arrival in Pasadena was to buy a car. That was not as easy as it sounds. The used car lots in Pasadena are sprinkled down Colorado Boulevard for several miles in typical US fashion, and getting to them in the days when public transport in Los Angeles was probably at its lowest ebb was not straightforward. It was only after my wife and I were stopped by the police and asked why we were walking on the streets of Pasadena that I understood the paradox that, in California, you had to have a car to buy a car. Another chicken-and-egg problem arose in connection with "ID," a term we had not encountered before. As a matter of routine, the police demanded to see our ID and of course the only acceptable ID in deepest Pasadena at that time was a California driver's license. A British driving license without a photograph of the bearer was clearly inadequate, and even our passports were looked on with suspicion.

An introduction to America via used car salesmen is not the introduction I would recommend to my worst enemy, and it is not surprising that I sought advice from the Caltech grad students. I was pointed in the direction of Steve Ellis, whose advice was valued because he came from Detroit and was believed to be worldly-wise. I tracked Steve down to the seminar room, where I saw he was engaged in a debate with a character who looked mildly reminiscent of the used car salesmen I had recently encountered. That was, of course, my first introduction to Dick Feynman. At first, I did not recognize him from the much earlier photograph I knew from the three red books of the *Feynman Lectures on Physics* (Addison-Wesley, 1963). Curiously enough, even after ten years or more, I always felt more comfortable addressing him as Feynman rather than Dick.

No doodling in science

Compared to my previous life as a graduate student in Oxford, life at Caltech was like changing to the fast lane on a freeway. First, instead of Oxford being the center of the universe, it was evident that, to a first approximation, Europe and the UK did not exist. Second, I rapidly discovered that the ethos of the theory group of Feynman and Gell-Mann was that physics was all about attacking the outstanding fundamental problems of the day: It was not about getting the phase conventions right in a difficult but ultimately well understood area. I remember asking George Zweig, a coinventor of the whole quark picture of matter, for his comments on a paper of mine. It was the not-about-to-be-very-famous SLAC-PUB 1000, a paper I had written with an experimenter friend at the Stanford Linear Accelerator Center (SLAC) about the analysis of three-body final states. George's uncharacteristically gentle comment to me was, "We do, after all, understand rotational invariance." In fact, the paper was both useful and correct but, on the Caltech scale of things, it amounted to doodling in the margins of science. In those days, I aspired to be as good a physicist as Zweig: This ambition strikes me now as similar to wanting to emulate the achievements of Jordan in the early days of quantum mechanics, rather than those of his collaborators, Heisenberg and Born.

One of the nicest things about Caltech was the sheer excitement of being around Feynman and Gell-Mann. As a postdoc from England, where one gains a rapid but narrow exposure to research, my wife and I were contemporary in age with the final-year grad students, and a lot of our social life was spent with them. Feynman was actively working with two of them, Finn Ravndal and Mark Kislinger, who had just been awarded his PhD for

TONY HEY is the chair of the electronics and computer science department at Southampton University in the United Kingdom. He is also the editor of The Feynman Lectures on Computation, *scheduled for publication this month. This article is adapted from the "Afterword" in that book, ©1996 by Anthony J. G. Hey, with permission of Addison-Wesley Publishing Company Inc. All rights reserved.*

44 SEPTEMBER 1996 PHYSICS TODAY

© 1996 American Institute of Physics, [illegible]

Keystoning and baselines (1)

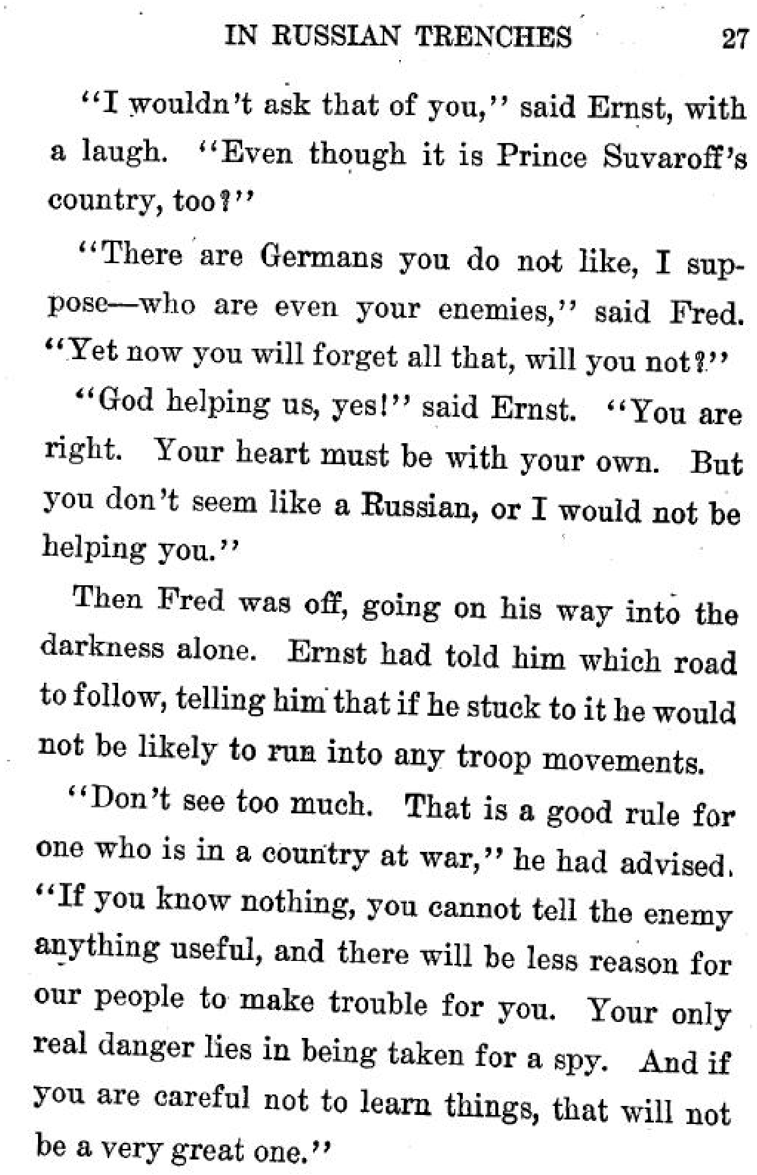
- `Pix *pix = pixDeskewLocal("keystone.png", 10, 0, 0, 0.0, 0.0, 0.0);`

  Find local skew in horizontal slices
  Fit the skew(y) to a straight line
  Compute the 8-pt projective transform
  Deskew using the transform

- `Numa *na = pixFindBaselines(pix, &pta);`

  The Numa gives the baseline (y) for each textline
  The Pta gives left and right ends of each textline
  These are used to display the baselines

Keystoning and baselines (2)



IN RUSSIAN TRENCHES 27

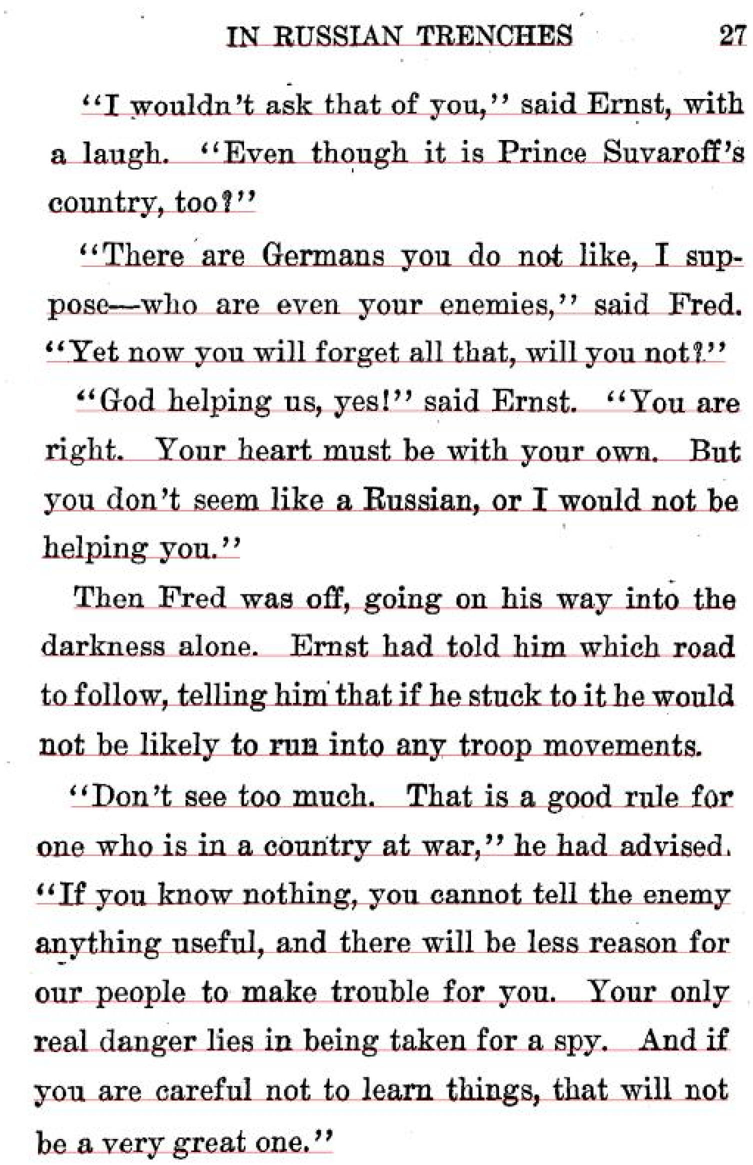
"I wouldn't ask that of you," said Ernst, with a laugh. "Even though it is Prince Suvaroff's country, too?"

"There are Germans you do not like, I suppose—who are even your enemies," said Fred. "Yet now you will forget all that, will you not?"

"God helping us, yes!" said Ernst. "You are right. Your heart must be with your own. But you don't seem like a Russian, or I would not be helping you."

Then Fred was off, going on his way into the darkness alone. Ernst had told him which road to follow, telling him that if he stuck to it he would not be likely to run into any troop movements.

"Don't see too much. That is a good rule for one who is in a country at war," he had advised. "If you know nothing, you cannot tell the enemy anything useful, and there will be less reason for our people to make trouble for you. Your only real danger lies in being taken for a spy. And if you are careful not to learn things, that will not be a very great one."



IN RUSSIAN TRENCHES 27

"I wouldn't ask that of you," said Ernst, with a laugh. "Even though it is Prince Suvaroff's country, too?"

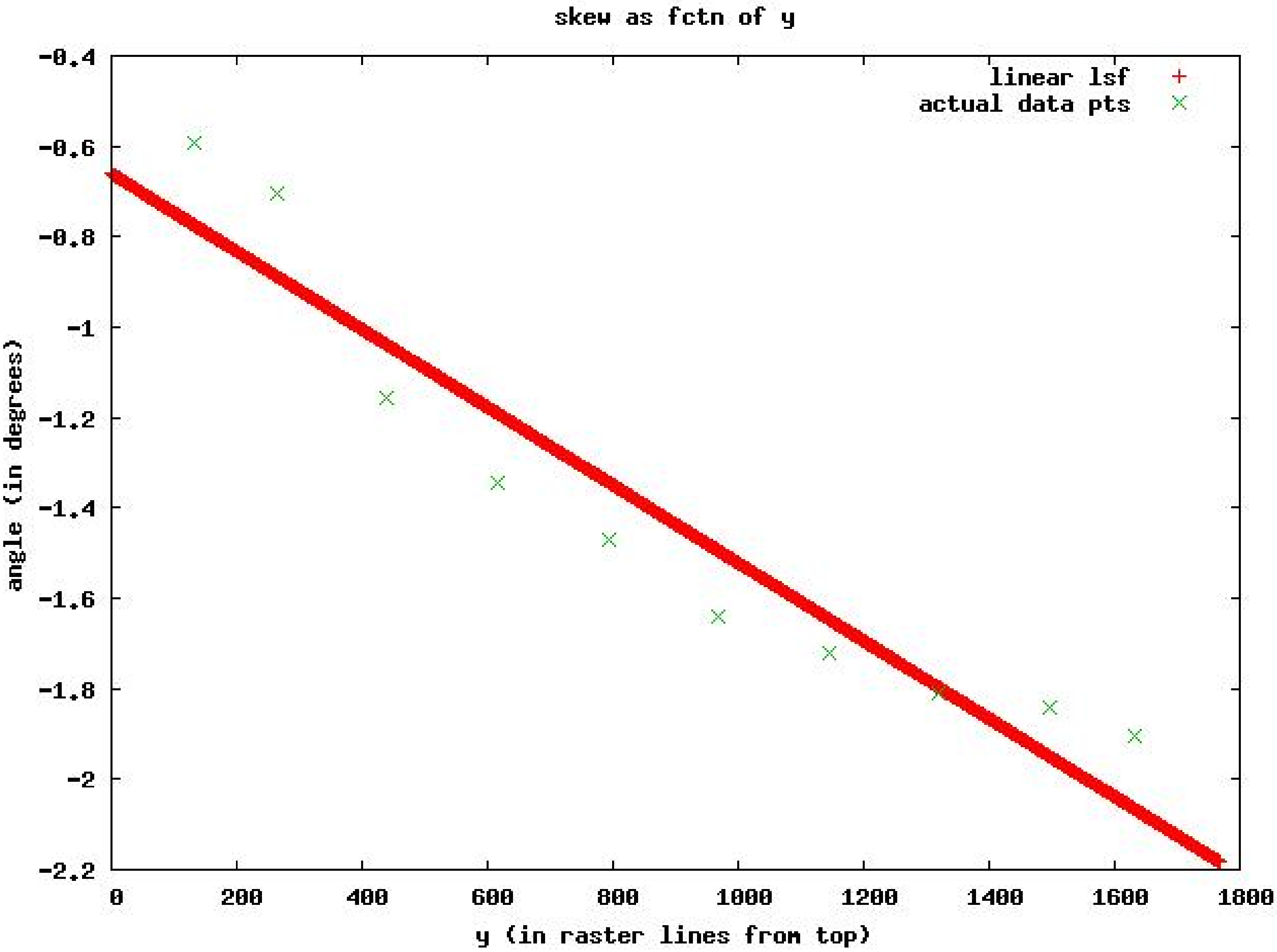
"There are Germans you do not like, I suppose—who are even your enemies," said Fred. "Yet now you will forget all that, will you not?"

"God helping us, yes!" said Ernst. "You are right. Your heart must be with your own. But you don't seem like a Russian, or I would not be helping you."

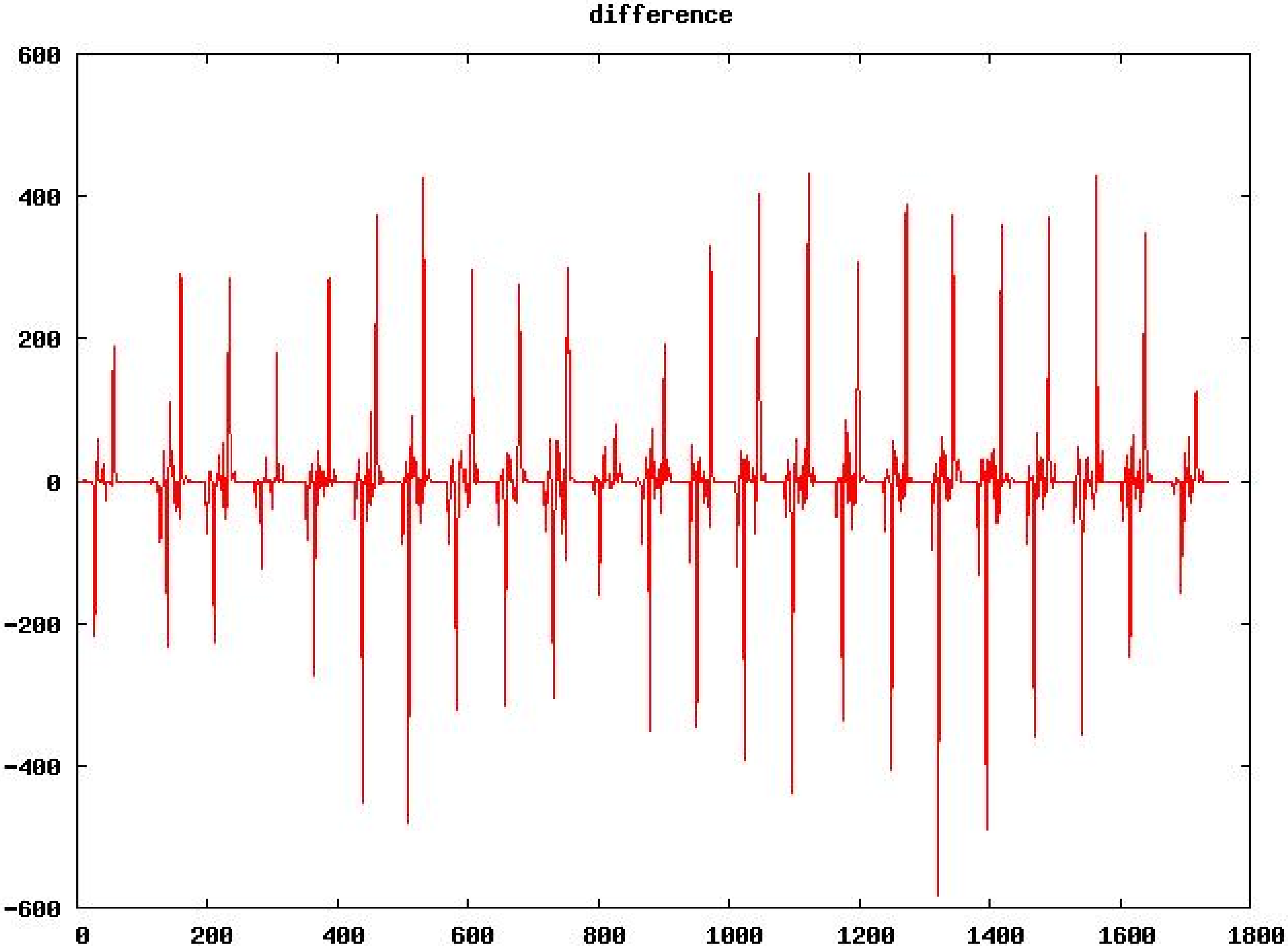
Then Fred was off, going on his way into the darkness alone. Ernst had told him which road to follow, telling him that if he stuck to it he would not be likely to run into any troop movements.

"Don't see too much. That is a good rule for one who is in a country at war," he had advised. "If you know nothing, you cannot tell the enemy anything useful, and there will be less reason for our people to make trouble for you. Your only real danger lies in being taken for a spy. And if you are careful not to learn things, that will not be a very great one."

Keystoning and baselines (3)



Keystoning and baselines (4)



Unsupervised shape classification

- General reference: *www.leptonica.org/jbig2.html*
- Identifies connected components (e.g., characters) in 1 bpp images
- Places them in equivalence classes
- Can also make classes of words (e.g., *dimsum*)
- Can use either correlation or rank hausdorff for decision
- Aggregates components over multiple pages
- This is used in Adam Langley's JBIG2 open source encoder

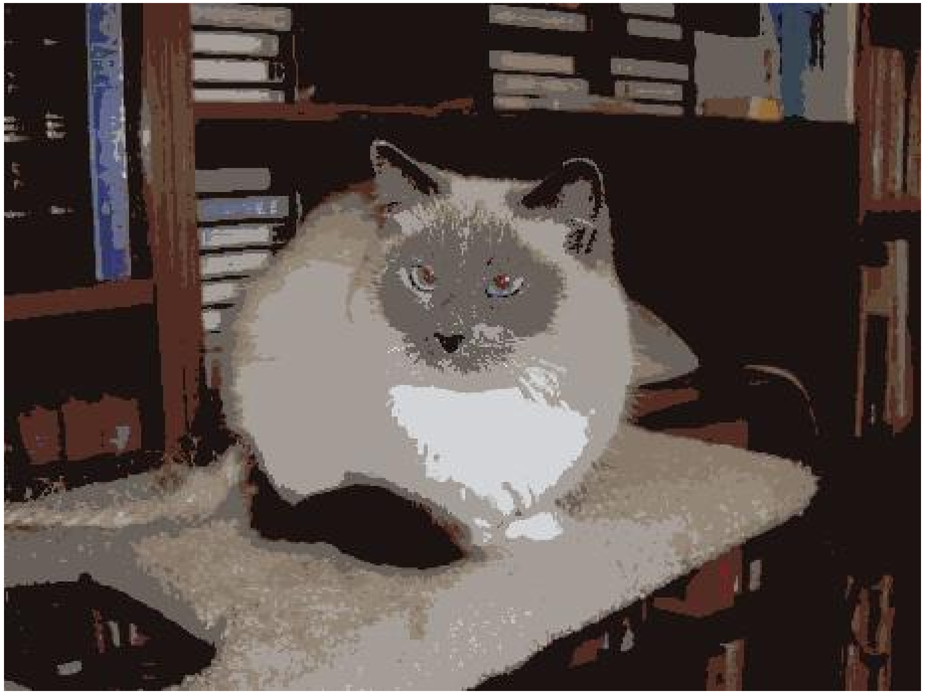
  www.imperialviolet.org/jbig2.html
- Must be careful with baselines
- The JBIG2 encoder was used to generate PDFs for Google Book Search

  www.leptonica.org/papers/google-books-pdf.pdf

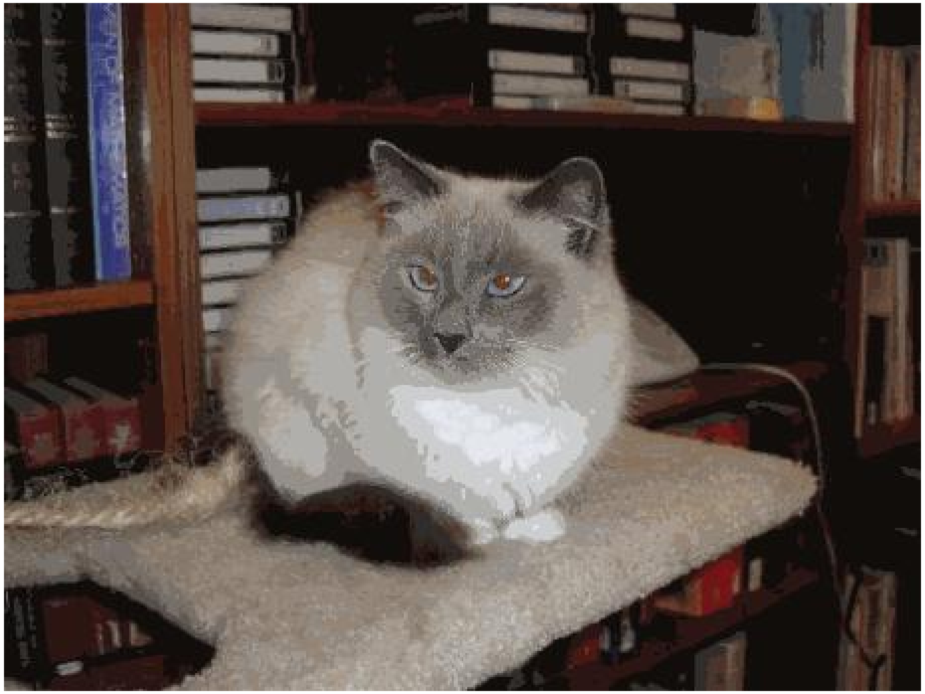
Color quantization and color segmentation (1)

- Why color quantization?
  - Need few levels for text
  - Better compression
  - Impressionist artwork
  - Can use for color seg.
- Octcube is efficient method
  - Populate at different depths
  - Fast lookup for quantization
- Dither for rendering accuracy (not MSE)
- Generating a colormap vs. quantizing to a colormap

Color quantization and color segmentation (2)

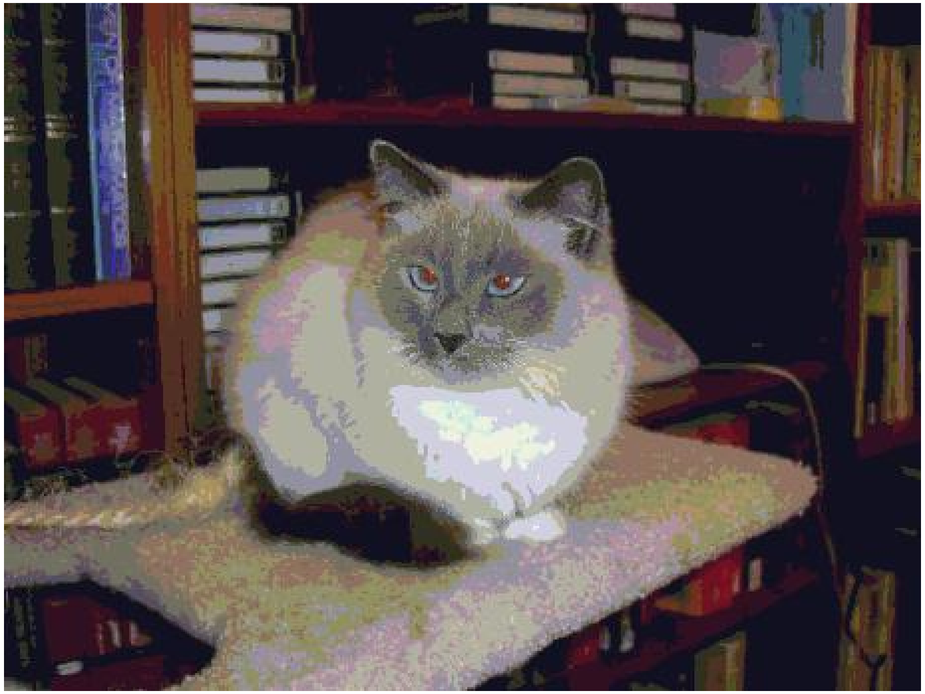


- Fixed levels; depth 2
- 27 colors

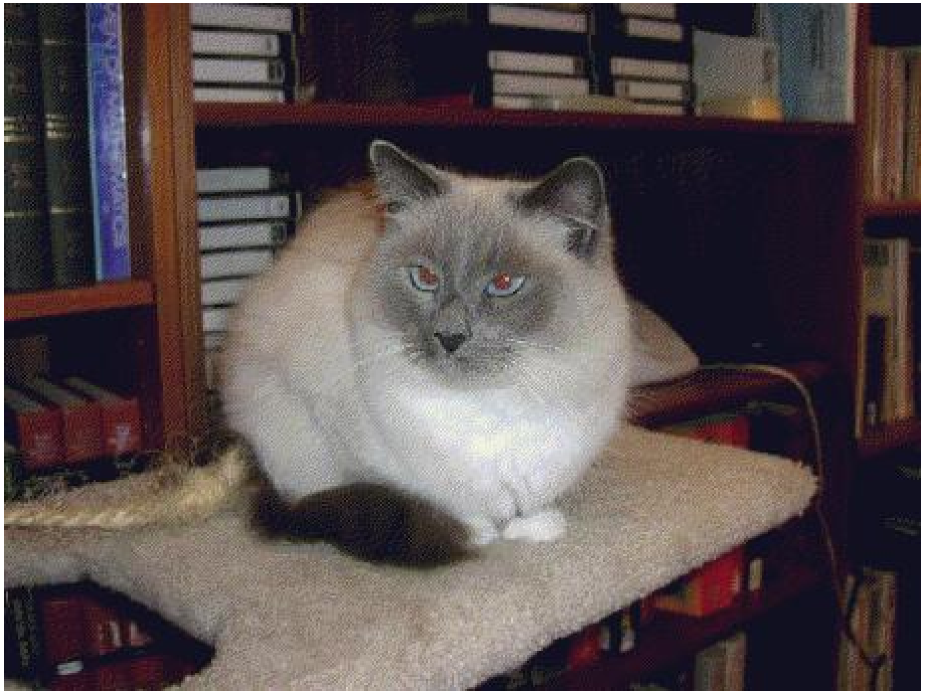


- Fixed levels; depth 3
- 86 colors

Color quantization and color segmentation (3)

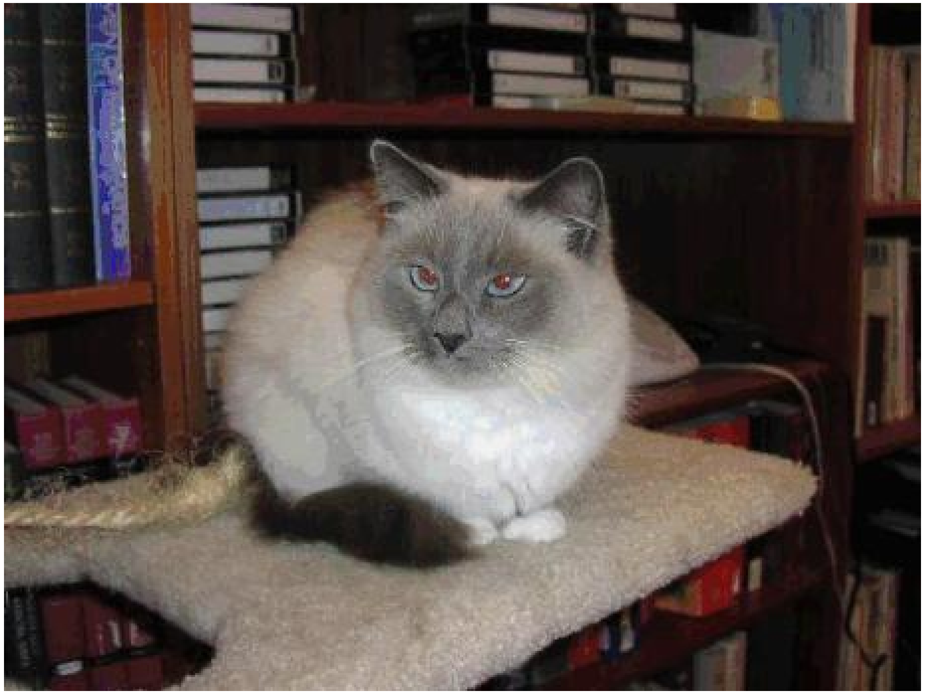


- 256 cells (3,3,2); no dithering
- 56 colors

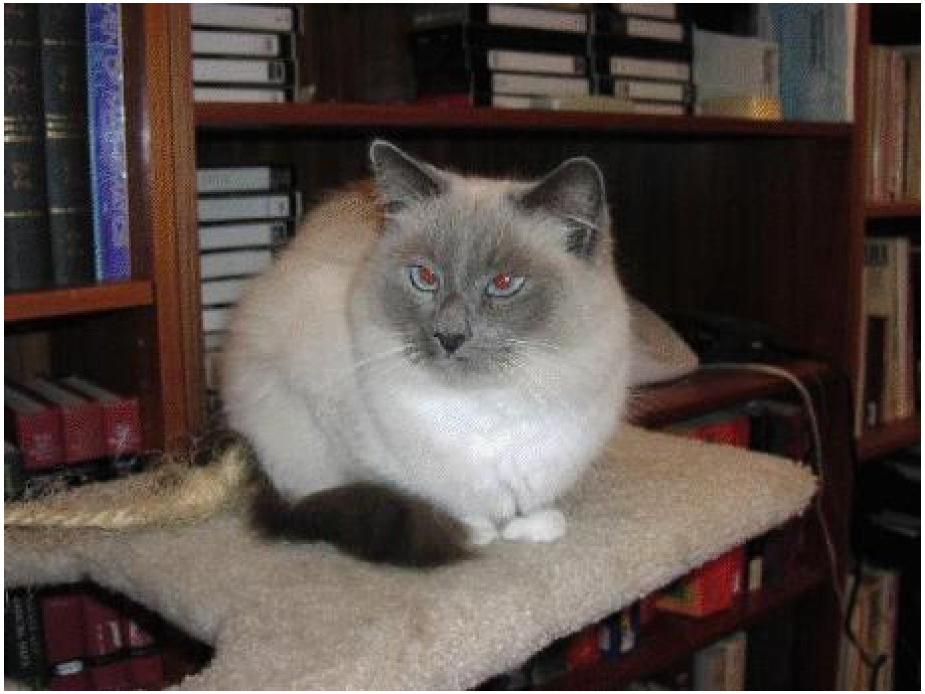


- 256 cells (3,3,2); dithered
- 81 colors

Color quantization and color segmentation (4)

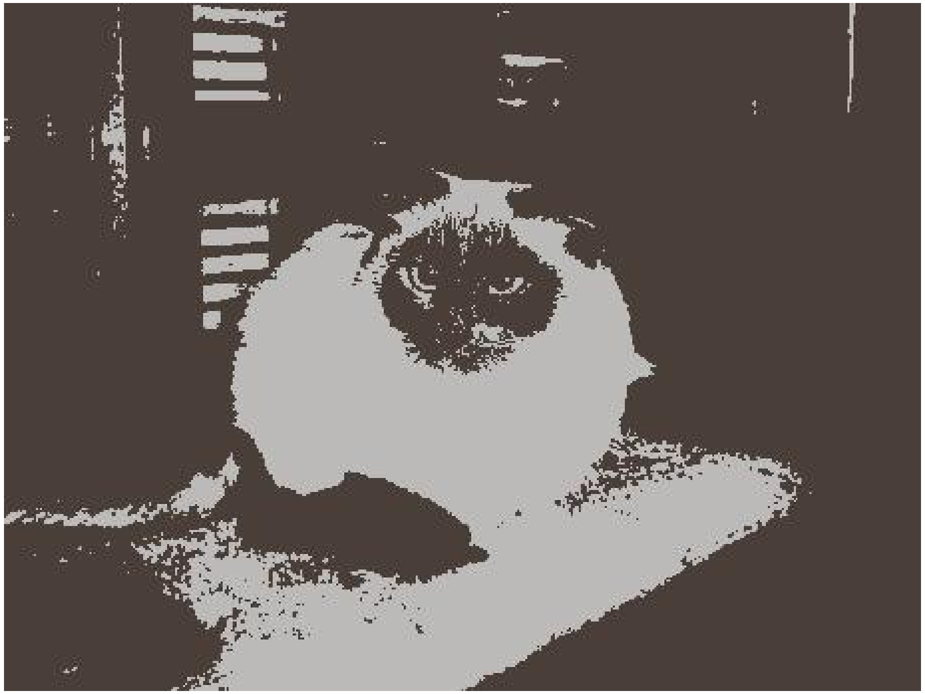


- 2-pass octree; no dithering
- 174 colors

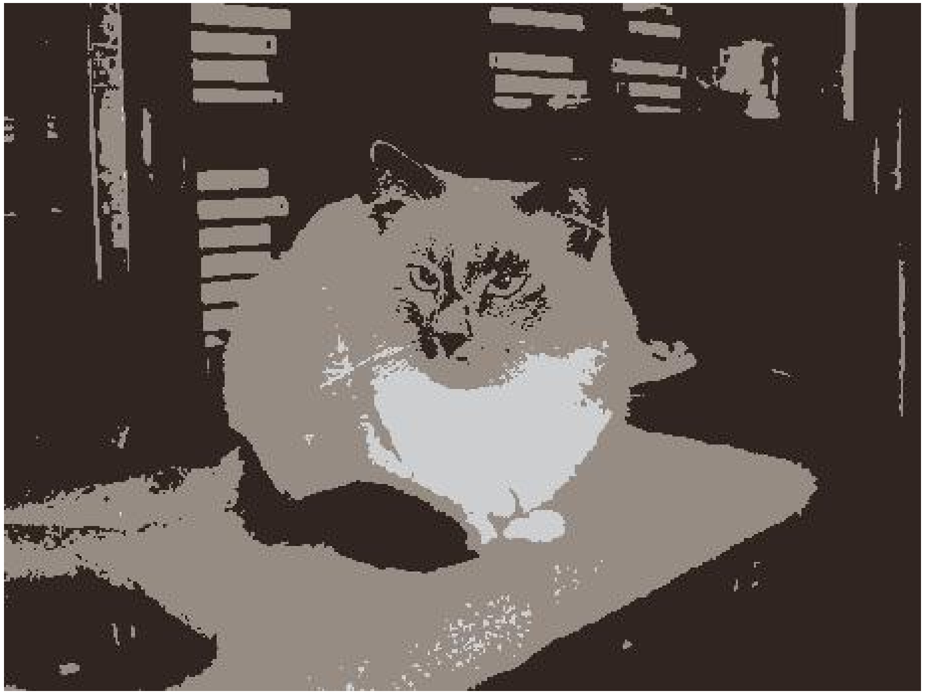


- 2-pass octree; dithered
- 190 colors

Color quantization and color segmentation (5)

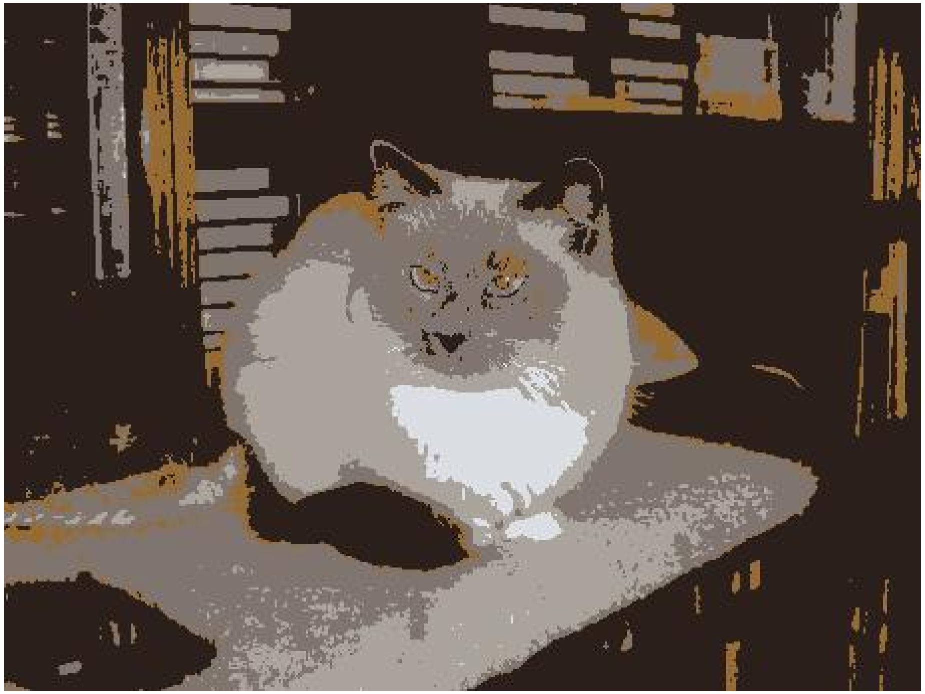


- color segmentation
- 2 colors

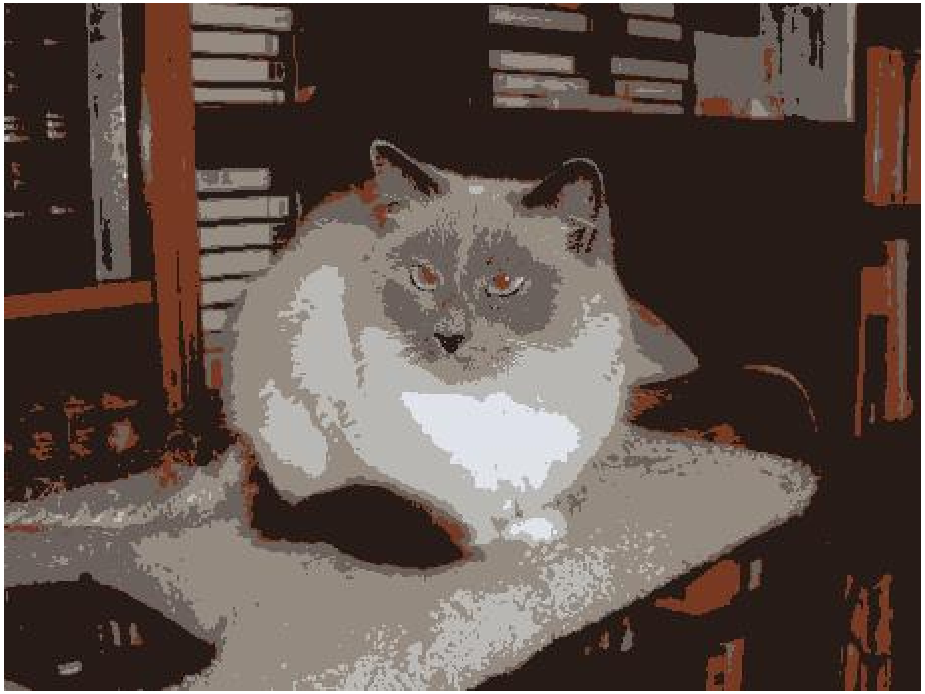


- color segmentation
- 3 colors

Color quantization and color segmentation (6)



- color segmentation
- 5 colors



- color segmentation
- 6 colors

Leptonica library extras

- Programmatic interface to gnuplot
- Simple bitmap font facility
- Blending images and simple line graphics
- Generating outlines from rasters and raster conversion from outlines
- Number and string arrays, heaps, stacks, queues, lists, etc.
- Octree color quantization
- Parser to extract C prototypes for a header file
- A large number of regression tests and example programs.