IJDAR (2005)
Digital Object Identifier (DOI) 10.1007/s10032-004-0138-z

International Journal on IJDAR Document Analysis and Recognition

Camera-based analysis of text and documents: a survey

Jian Liang, David Doermann, Huiping Li

Language and Media Processing Laboratory, Institute for Advanced Computer Studies, University of Maryland, College Park, USA (e-mail: {lj,doermann,huiping}@umiacs.umd.edu)

Received: 18 December 2003 / Accepted: 1 November 2004
Published online: ♣ 2005 – © Springer-Verlag 2005

Abstract. The increasing availability of high-performance, low-priced, portable digital imaging devices has created a tremendous opportunity for supplementing traditional scanning for document image acquisition. Digital cameras attached to cellular phones, PDAs, or wearable computers, and standalone image or video devices are highly mobile and easy to use; they can capture images of thick books, historical manuscripts too fragile to touch, and text in scenes, making them much more versatile than desktop scanners. Should robust solutions to the analysis of documents captured with such devices become available, there will clearly be a demand in many domains. Traditional scanner-based document analysis techniques provide us with a good reference and starting point, but they cannot be used directly on camera-captured images. Camera-captured images can suffer from low resolution, blur, and perspective distortion, as well as complex layout and interaction of the content and background. In this paper we present a survey of application domains, technical challenges, and solutions for the analysis of documents captured by digital cameras. We begin by describing typical imaging devices and the imaging process. We discuss document analysis from a single camera-captured image as well as multiple frames and highlight some sample applications under development and feasible ideas for future development.

1 Introduction

Document image analysis has carved a niche out of the more general problem of computer vision because of its pseudo binary nature and the regularity of the patterns used as a "visual" representation of language. In the early 1960s, optical character recognition was taken as one of the first clear applications of pattern recognition, and today, for some simple tasks with clean and well-formed data, document analysis is viewed as a solved problem. Unfortunately, these simple tasks do not represent the most common needs of the users of document image analysis. The challenges of complex content and layout, noisy data, and variations in font and style presentation keep the field active.

Document image processing and understanding has been extensively studied over the past 40ỹears. Work in the field covers many different areas including preprocessing, physical and logical layout analysis, optical and intelligent character recognition (OCR/ICR), graphics analysis, form processing, signature verification, and writer identification and has been applied in numerous domains, including office automation, forensics, and digital libraries. Some surveys include [19,72,81,93].

Traditionally, document images are scanned from pseudo binary hardcopy paper manuscripts with a flatbed, sheet-fed, or mounted imaging device. Recently, however, the community has seen an increased interest in adapting digital cameras to tasks related to document image analysis. Digital camcorders, digital cameras, PC-cams, PDAs, and even cellphone cameras are becoming increasingly popular, and they have shown potential as alternative imaging devices. Although they cannot replace scanners, they are small, light, easily integrated with various networks, and more suitable for many document capturing tasks in less constrained environments. These advantages are leading to a natural extension of the document processing community where cameras are used to image hardcopy documents or natural scenes containing textual content.

The industry has sensed this direction and is shifting some of the scanner-based OCR applications onto new platforms. For example, XEROX's Desktop PC-cam OCR suite [75], based on their CamWorks project, aims to replace scanners with PC-cams in light-workload environments. The DigitalDesk project [96] turns the desktop into a digital working area through the use of cameras and projectors. The system can follow a pen tip or fingertip to select an area on a document and recognize the selected printed or handwritten symbols. Other applications are being enabled as well, such as intelligent digital cameras to recognize and translate signs written in foreign languages [95,105,106]. Current research is focused on processing single images of text, and although

video files are of much lower resolution and require more storage, technological and computational advances will make processing text from video an obtainable goal in the reasonable future. Although many of these technologies are in the research stage and not available commercially, substantial progress is being made and reported.

In this paper we provide a survey for recent work on camera-based document processing and analysis. In Sect. 2, we introduce the background and motivation for camera-based document analysis primarily for anyone with fundamental image processing knowledge but little specific knowledge of document analysis. It includes the introduction of various imaging devices and a discussion of the advantages, applications, and challenges of camera-based document analysis. Section 3 concerns the technologies for camera-based image acquisition. Section 4 provides technical details on processing single images of documents and text. There we elaborate various topics of text detection, extraction, enhancement, and recognition. Section 5 addresses processing text in video. We discuss three problems, i.e., key document frame selection, text tracking, and multiframe enhancement. Section 6 highlights some performance evaluation issues including a discussion of evaluation tools and metrics specifically designed for camera-based document image analysis. Although our attention is on document analysis, we do not rule out literature and applications on scene and graphic text analysis in images and video because they are seen as the first deployable applications.

2 Background

2.1 Text and documents

Work on the general problem of camera-based text and document analysis can be categorized in a number of ways: by the type of text processed, by the applied technology, by the intended application, or simply by the type of devices used. Generally, we can regard any scene that has textual content as a document, including video frames with captions, a vehicle and its license plate, or an opened book. The difference in the composition of the image, however, will define the challenges of extracting the text content. A majority of the work on camera-captured data has been done in the area of processing *image and video text* from broadcast video or still images, rather than on processing *images of structured documents*. Each problem has its unique challenges, but all are directed toward the ultimate goal of providing cameras with reading capabilities. We choose to organize the survey at the highest level as processing single images of traditional documents vs. multiframe content, typically containing scene and graphic text. We define these two domains briefly in what follows and provide more details in Sects. 4 and 5, respectively.

The problem of imaging and processing structured documents is mainly constrained by the use of scanners. This is a very well-studied realm that can be categorized under single document image processing in Sect. 4. The sources of images in this case are usually paper-based printed or handwritten documents, such as journal papers, business letters, faxes and memos, forms, and checks. A substantial part of the document is assumed to be text, while figures and pictures are allowed. Colorful documents are accepted by some of the recent technologies [90]. In this survey, our focus is on recent work that aims to replace the scanner with a more flexible camera device in appropriate situations. This problem has been less widely addressed.

The processing of image and video text is a specific application that seeks to recognize text appearing as part of, or embedded in, visual content. One branch of the previous work in the literature is concerned with text detection in video key frames (or still images), which has received a great deal of attention as it provides a supplemental way of indexing images or video. The community typically distinguishes between *graphic text*, which is superimposed on the image (such as subtitles, sports scores, or movie credits), and *scene text* (such as on signs, buildings, vehicles, name tags, or even T-shirts). The general goal is to provide the capability to capture information intended for visual human communication and use it for various navigation and indexing tasks. Similarly, reading image text has been widely addressed in the WWW community because of the desire to index text that appears in graphical images of many Web pages [59]. The problem presents many of the same challenges as reading text in general scene images but is primarily constrained to text that has been graphically overlaid on images.

Different documents and scene text require different devices to convert them into digital format. In what follows we will discuss various imaging devices for this purpose.

2.2 Imaging devices

2.2.1 Production devices. Digital scanners have been the dominant imaging devices for documents in the past decade, and many new types of scanners have been developed for various imaging tasks. Scanners range from large-format drum scanners used for engineering drawings to small desktop devices used for casual scanning of business cards. The speed of scanners can be as high as several pages per second with sheet-fed sources or as low as a second per line for low-end devices that require manually moving a wand one word at a time. With the help of robot arms, it is even possible to automatically turn pages of bound books. The resolution of consumer-grade flatbed scanners has recently passed 2400 dpi (dots per inch), and those for film scanning can be much higher, and at the same time the price of consumer-grade scanners has fallen well below $100, making them very popular PC add-ons. Such scanners are more than adequate for document image analysis, given that 300 dpi is the standard in OCR. Since scanners work adequately for acquiring hardcopy pages as digital images, they are not typically considered as a significant problem in the process of document analysis.

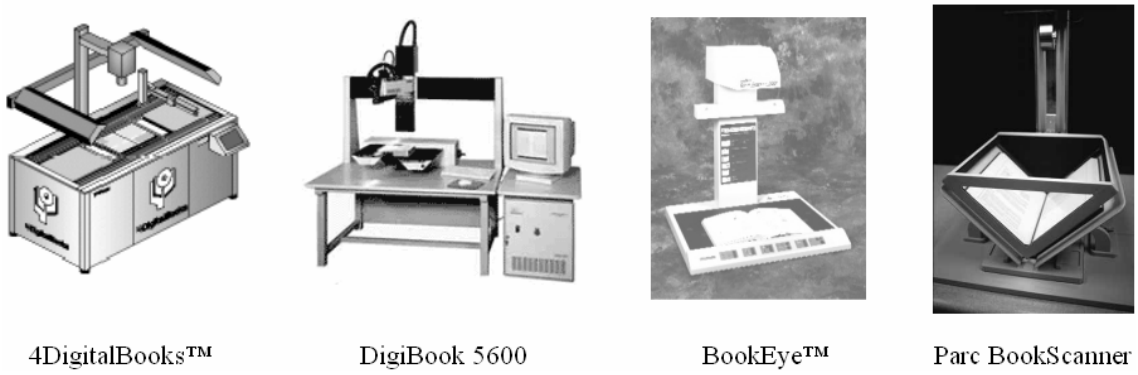

Fig. 1. Industrial cameras for document image capturing

Cameras are also widely used in document image analysis, though much less prevalent in the literature. These cameras are mostly industrial-grade – high quality and expensive, as are the systems that use them. Sometimes they are called planetary cameras/scanners. Typically, a special environment is set up to keep the manuscript stationary, as flat as possible (possibly under a glass), and well lit. Often the camera moves along a high-precision rack in order to capture large size documents. The main reason for using a high-end camera is to deal with material that cannot be scanned, like thick rare books, fragile historical manuscripts, or brittle paper, so the working environment created is as ideal as possible. Some planetary cameras are accompanied by smart software that can even restore the distorted text lines near book spines (e.g., BookEye). Recently, many companies have begun to downsize this idea for the desktop environment [126]. Some are used to replace transparency projectors with digital projectors, while others are used to replace scanners for convenient image capture. Figure 1 shows some representative industrial-level cameras for book scanning, ranging from a large frame robotic machine to desktop setup.

2.2.2 Digital cameras. The booming of consumer-grade digital cameras in the past 5 years has brought a large number of imaging devices into the hands of ordinary people who, until recently, had little demand for a scanner. This trend will continue in the near future with the expansion of cameras into mobile phones. The most important feature of these cameras is their flexibility. They can be as small as a business card, weatherproof, carried anywhere, and easily used. Likewise, the border between image- and video-capture devices is disappearing with integrated digital cameras and camcorders. As a result, the documents they may want to capture will not necessarily be fixed, flat, or well lit, and the digital copy may be either still images or video sequences. When high-quality devices are replaced by devices meant for daily life, these flexible working conditions introduce new levels of image processing demands that need to be addressed. The fact that the environments digital cameras are operating in are no longer as constrained as that of scanner or industrial cameras presents new challenges.

Current consumer-grade digital cameras are expanding to 8 megapixels and beyond, with resolutions of up to 3500×2200. Under ideal imaging conditions, this should be sufficient for capturing standard size documents at a resolution (300 dpi) adequate for document image analysis. Although most of the devices are still in the 2- to 3-megapixel range, in the next several years these higher resolution devices will likely be affordable. For some applications, mass capture via video is appropriate. Current digital video cameras typically have much lower resolution (640×480) because they are designed primarily for low-bandwidth presentation and are often highly compressed. The fact that they are not designed specifically for document image capture presents many interesting challenges. Ultimately, we hope to be able to perform various document analysis tasks directly on the device. Recently we have seen consumer and business applications intended to eventually run on PDAs with cameras or even on cellular phones [118]. Many companies currently market compact flash cameras that can be attached to pocket or tablet PCs for document capture. Nokia and other telecom companies have recently released camera phones that capture at a resolution up to 640×480 with over 1 megapixel. Although not yet sufficient for capturing full documents, these resolutions have been shown to be sufficient for scene text. Figure 2 gives an idea of the resolution and price ranges of most consumer-grade digital cameras that may be the frontend of future document analysis systems.

2.3 Advantages of camera acquisition

Document analysis using cameras has a number of advantages over scanner-based input. Cameras are small, easy to carry, and easy to use. They can be used in any environment and can be used to image documents that are difficult to scan such as newspapers and books or

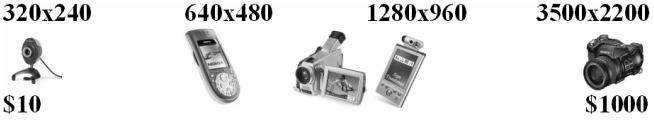

Fig. 2. Resolution and price ranges of consumer-grade digital cameras

Table 1. Comparison of scanners and digital cameras (*bold cells* represent devices with an advantage)

	Scanners	Digital cameras
Resolution	**150–600dpi**	50–300dpi
Object surface	**Flat (almost always)**	Can be arbitrary
Distortion	**Minimal for flat pages**	Perspective and optical lens
Illumination	**Even and adequate**	Difficult to control
Background	**Domain dependent / known**	Often complex
Blur	**Minimal**	Motion and out-of-focus
Batch speed	**Fast**	Slow
One-shot speed	Slow	**Fast**
Zoom / focus	Fixed	**Variable**
Portability	Bad	**Good**
Usability	Low	**High**
Size	Large	**Compact**

text that does not typically originate as hardcopy, such as text on buildings, vehicles, or other objects moving in a scene. In general, camera-based systems are more flexible.

A user study conducted by Newman et al. [75] shows that desktop OCR using PC-cams is more productive than a scanner-based OCR for extracting text paragraphs from newspapers. Fisher [27] investigates the possibility of replacing sheet-fed scanners used by soldiers in the battlefield, with digital cameras. He finds that sheet-fed scanners cannot be used to capture thick books, and they are bulky and difficult to maintain, making them unsuitable for battlefields. Experimentation leads him to the conclusion that digital cameras are capable of capturing a whole A4 size document page at an equivalent 200 dpi resolution needed by OCR. Table 1 gives a brief comparison of scanners vs. digital cameras.

2.4 Challenges

State-of-the-art document analysis software can produce good results from clean documents. However, the techniques assume high-resolution, high-quality document images with a fairly simple structure (black text on a white background). Unfortunately, these assumptions are not typically valid for camera-based systems. The major challenges include:

Low resolution. – Images taken with cameras usually have a low resolution. While most OCR engines are tuned to resolutions between 150 and 400 dpi, the same text in a video frame may be at or below 50 dpi, making even simple tasks such as segmentation difficult.

Uneven lighting. – A camera has far less control of lighting conditions on an object than scanners. Uneven lighting is common, due to both the physical environment (shadows, reflection, fluorescents) and uneven response from the devices. Further complications occur when trying to use artificial light or due to image reflective surfaces. As Fisher [27] found, if on-camera flash is used, the center of the view is the brightest, and then lighting decays outward.

Perspective distortion. – Perspective distortion can occur when the text plane is not parallel to the imaging plane. The effect is that characters farther away look smaller and are distorted and parallel-line assumptions no longer hold in the image. The ultimate effect on page segmentation and OCR depends on the specific algorithm. From our experience we find that small to mild distortion may cause significant trouble for some commercial OCR packages. For flatbed scanners, documents are perfectly aligned with the scanning surface, so translation and rotation are the primary challenges.

Nonplaner surfaces. – Scene text can appear on any surface, not necessarily on a plane. Pages of an opened book are rarely flat and are more often curled. All these nonplaner surface cases will cause trouble for current document analysis tools that are tailored for scanner-based applications. Like perspective distortion, even moderate warping can cause most current OCR systems to fail.

Wide-angle-lens distortion. – As an imaged object gets closer to the image plane, lighting, focus, and layout distortions often occur on the periphery. Since many focus-free and digital cameras come with a cheap wide-angle lens, distortion can be a problem if they are used for document analysis, although the distortion can be empirically modeled as a polynomial radial distortion function and restored to some extent.

Complex backgrounds. – Often more of the scene is imaged than the intended text or document. If the document we are imaging does not have a regular shape, it may be difficult to segment. The lack of a uniform background (even as simple as the background on a sign) can make segmentation especially difficult.

Zooming and focusing. – Since many digital devices are designed to operate over a variety of distances, focus becomes a significant factor. Sharp edge response is required for the best character segmentation and recognition. At short distances and large apertures, even slight perspective changes can cause uneven focus.

Moving objects. – The nature of mobile devices suggests that either the device or the target may be moving. The amount of light the CCD can accept is fixed, and at higher resolutions the amount each pixel gets is smaller, so it is harder to maintain an optimal shutter speed, resulting in motion blur. In the simplest case, motion blur can be modeled by a point spread function (PSF) that is tuned to the directions of motion. A more complex model is needed, however, when multiple motion directions are involved, or when objects are at different distances from the camera.

Intensity and color quantization. – In an ideal imaging device, each pixel in a photon sensor (CCD/CMOS) array should output the luminance of the inbound light and/or color components (RGB or YUV) corresponding to the frequency of the light. In practice, however, different hardware designs have different spatial/intensity/color quantization mechanisms. The first issue is the low-pass filter used in many digital cameras. Most current CCD/CMOS- based cameras place the RGB sensors in a Bayer format [2]. This pattern has twice as many G sensors as R and B sensors. Each pixel can only see one color. A low-pass filter is applied, usually in hardware, to spread the color to nearby positions so as, e.g., to generate our familiar RGB output at each pixel. By contrast, most scanners use separate CCD/CMOS sensors for RGB components that are spatially separate and therefore do not have this low-pass filter and may produce sharper images. The second issue is related to the photon sensor size. Since scanners rely on moving light and optics to scan, they do not have to squeeze as many pixels as digital cameras on a single CCD/CMOS chip. The larger photon sensor size results in a better dynamic range. Current digital cameras can easily under-/overexpose due to their small photon sensor size on a crowded CCD/CMOS chip.

Sensor noise. – Dark noise and read-out noise are the two major sources of noise at the CCD/CMOS stage in digital cameras. Additional noise can be generated in amplifiers. The higher the shutter speed, the smaller the aperture, the darker the scene, and the higher the temperature, the greater the noise. Compared to digital cameras, scanners normally have less to worry in all these aspects.

Compression. – Most images captured by digital cameras are compressed. It is possible to obtain uncompressed images at the cost of five- to tenfold storage space. Due to the limited resources in mobile applications, they usually rely heavily on compression. However, current compression schemes are optimized, not for document analysis, but for general scene images, which creates challenges for document image analysis for which it is necessary to preserve sharpness.

Lightweight algorithms. – The ultimate goal will be to embed document analysis processing directly into the devices. In such cases, the system must provide computationally efficient algorithms that can operate with limited memory, processor, and storage resources.

Figure 3 shows some of the challenges in camera-based document analysis. Note that the comparison of Fig. 3a.1 and b.1 clearly shows a much lower resolution of Fig. 3b compared to 3a. The uneven illumination is illustrated by Fig. b.2 and b.3, where the dark text in Fig. b.2 is brighter than the white space in Fig. b.3. No global thresholding binarization methods would be able to handle both at the same time. Also, notice the problem of out-of-focus blur in Fig. b.2 as compared to Fig. b.3. Similar problems can be found in Fig. c, too, while nonplanar warping presents some more difficulties.

2.5 Camera-based applications

Over the past 20 years, there have been numerous applications on camera-based text recognition, such as reading license plates, book sorting [31], visual classification of magazines and books, reading freight train IDs, road sign recognition, detection of danger labels, and reading signs in warehouses. Figure 4 shows some example applications. In addition to these types of applications, the ability to process signs using mobile, low-cost hardware enables numerous other applications.

License plate reading. – With the rapid deployment of traffic control systems, automatic license plate reading (ALPR) is playing an important role in many applications. Many systems [8,16,117] have been developed for ALPR and many commercial products put to practical use [119–121] in parking lot billing, toll collecting monitoring, road law enforcement, and security management. ALPR systems have two significant characteristics. First, constraints on license numbers provide very useful cues for localization and segmentation. Second, however, it is difficult to distinguish certain numbers and symbols, such as “B” vs. “8” and “D” vs. “O”.

Sign detection and translation. – The ability to detect and recognize text using PDAs or cellular phones has promise for both commercial and military applications. Watanabe [95] was the first to describe a sign translation system (Japanese to English). Yang et al. [105,106] implement an experimental Chinese-to-English sign translation system and give a good analysis on the three components of a general sign translation system: sign detection, character recognition, and sign translation. They point out that the key difficulty is in the concise nature of signs: a sign is often comprised of only a few words/characters. In the image processing phase,

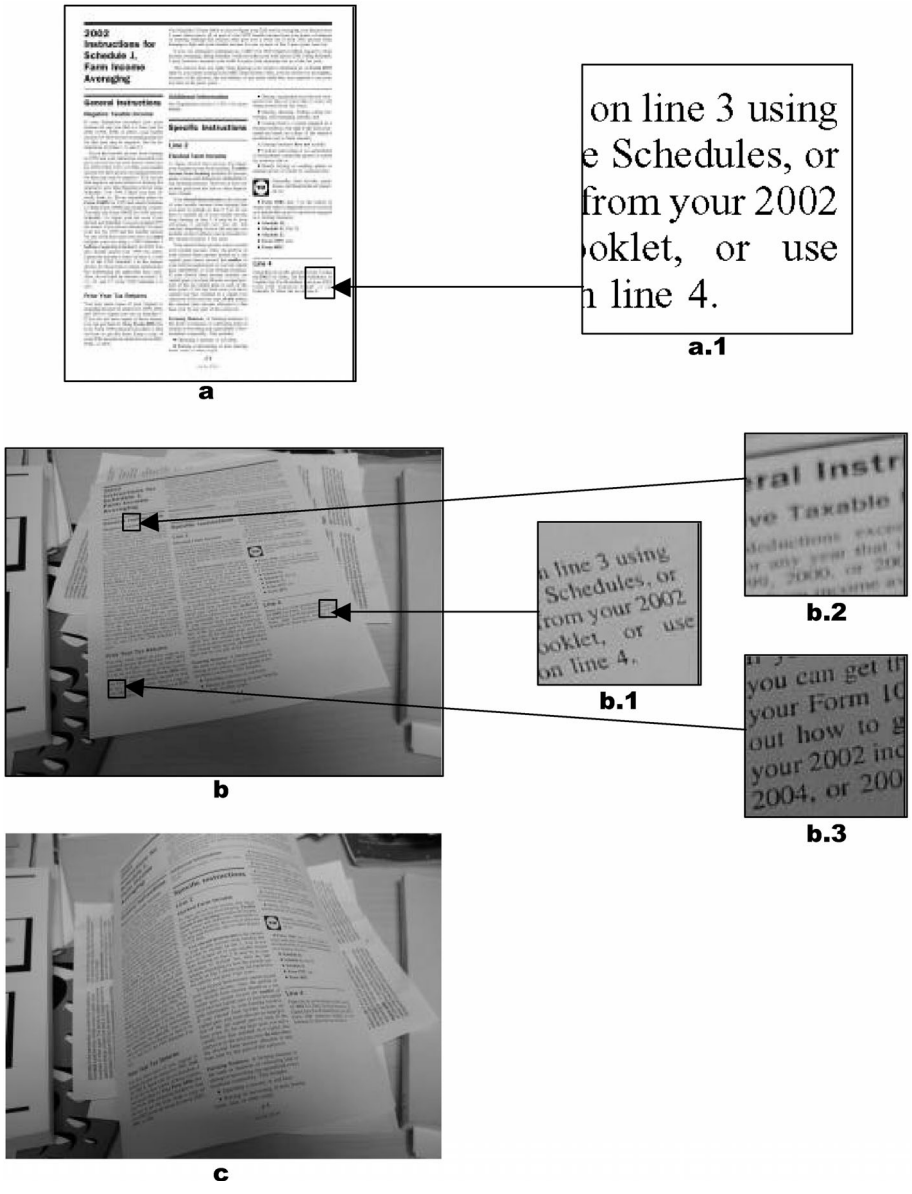

Fig. 3. Somme challenges for camera-based document analysis. **a** A document scanned at 300 dpi. **b,c** The same document captured by a digital camera, with perspective distortion, warping, and complex backgrounds. **a.1,b.1** The same small portion of the document captured by scanner and camera, respectively, both at full resolution. **b.2,b.3** Two small portions of **b** at full resolution

this will cause problems in sign detection and character recognition because the prevailing technologies are designed for large text segments. In the linguistic processing phase, this will cause problems in character recognition and sign translation because both OCR and machine translation (MT) need a certain amount of context for error correction to work. With limited context it is more difficult to distinguish different meanings of the same word. The authors find that traditional knowledge-based MT works well with grammatical sentences, but it has difficulty with ungrammatical text in signs.

Mobile text recognizer and speech generator for the visually impaired. – Zandifar et al. [107] propose to use camera-based OCR techniques in a head-mounted smart video camera system to help the visually impaired. Their goal is to detect and recognize text in the environment and then convert text to speech. The problems they confront on the vision side include the detection of text and the adjustment of cameras (such as zooming) so clear focus can be achieved. Other challenging problems include multiple text orientations and text on curved surfaces such as cans. On the audio side, one interesting problem would be how to generate meaningful speech based on erroneous recognition.

Cargo container and warehouse merchandise code reader. – Lee and Kankanhalli [49] present a system used in ports to automatically read cargo container codes. A single grayscale image captured by a camera is provided for reading container codes. The uneven surface may make text look warped. Their text detection is based on vertical edges found in the image and a verification stage uses domain knowledge that container codes are standardized in a fixed format: four letters followed by seven digits in one or two lines.

Visual input. – Camera-based handwriting (and human gesture) input provides yet another way to interact with computers. For example, a camera mounted over a writing surface can be used for handwriting recognition [26,70,96], or cameras can be used in whiteboard reading [89,97]. Without being forced to write on special writing boards, people will feel more comfortable and communicate more efficiently. Camera-based scene text recognition (and human gesture) understanding can also be used in offline video-based presentation analysis to catalog the recorded video and synchronize the slide files with video.

Document archiving. – High-end digital cameras have long been used in large-scale book digitizing projects [122,123]. Other lightweight desktop applications are also available [75,96]. As for consumer-grade equipment, due to their flexibility and independence of bulky computers, it will not be surprising to find digital cameras and camcorders being used as document digitizing and archiving devices in the future. A user can carry such a device conveniently anywhere and record interesting document pages instantly. A working prototype based on a PDA was presented by HP recently [118]. Although it has many restrictions [79], it showed great potential in its combination of mobile devices and cameras.

Text acquisition. – For small items it is also useful to have the mobile OCR ability. For example, while barcodes are widely used, they have the disadvantage of not being readable to humans and require expensive, specialized laser readers. A recent trend is to develop barcode readers based on PDAs and cameras [69]. The ability to capture and recognize text would be a further useful complement to barcode readers. Similarly, in the package delivery industry, it would be helpful to recognize addresses and automatically route them to an appropriate destination.

3 Camera-based acquisition

One advantage of using a camera instead of a scanner is that a camera makes it possible to acquire images at some distance from the target. By zooming to the area of interest, we introduce challenges for autofocus and zoom. Furthermore, due to the low resolution, it is often not possible to capture all the text in one frame while keeping a reasonable font size. Thus an image mosaic technique is needed to put pieces of text images together to form a large high-resolution image. Last but not least is the human machine interface design that will come after automatic image acquisition.

3.1 Autofocus and zooming

In [67], Mirmehdi et al. propose a simple approach to autozooming for general recognition problems. If the background around an object has low variance compared to the object, then the variance in the observation window can be used as an indicator to find the best zoom. In [107], Zandifar et al. discuss autofocusing problems in designing a text reading system for the visually impaired. They assume that the best focus is achieved when edges are the strongest in the image. The sum of the differences between neighboring pixels, the sum of gradient magnitude, or the sum of a Laplacian filter's output can be used as the overall edge strength measure. Focus is adjusted until the measurement is optimized. Sample images show that all three measurements presented similar judgments to human eyes. Mirmehdi et al. [68] describe a system that can automatically locate text documents in a zoom-out view and control the camera to pan, tilt, and zoom in to get a closer look at the document. Assuming the document is directly facing the camera so that there is no perspective distortion, the system segments text lines and estimates the average text line height to determine the zoom for optimal OCR. The entire document is divided into several pieces accordingly, and the camera captures each piece after panning, tilting, and zooming. The small pieces are put together by mosaicing to obtain a complete document image, which is sent to a commercial OCR package. The autofocusing and zooming problem is a very interesting one since it has direct application in robots.

3.2 Image mosaicing

Image registration and mosaicing are well-researched topics in image processing. Jung et al. [40] use mosaicing to put together long text strings that appear in multiple video frames into a panorama image. In the CamWorks project [75], mosaicing is used to put together the images of the upper and lower part of a document page. In [108], a desktop OCR system using a PC-cam is described where the camera is placed on top of a desk pointing downwards but the camera captures only a small part of an A4 document. The user moves the document while monitoring the computer screen until every part of the page appears in the sequence. During the capturing, frames are selected such that they are substantially different and yet successive ones overlap. This reduces the number of frames used in image registration and reduces blur that can result from the combination of too many images. Based on the observation that words are abundant in text documents, Zappala et al. adopt a feature-based image registration method, where feature points are the lower right vertices of word bounding boxes. The overlapping parts of two registered images are blended to avoid any abrupt seam. Pilu and Isgro [80] propos similar work in document mosaicing. Compared to general photo mosaicing, document analysis needs high resolution and high accuracy. The human eye is more sensitive to errors in documents consisting of linear structures than in general photos. In the above desktop applications, the documents are constrained in shape and position. Document mosaicing from handhelds, however, will need to address the problem of mosaicing under arbitrary shape and position conditions.

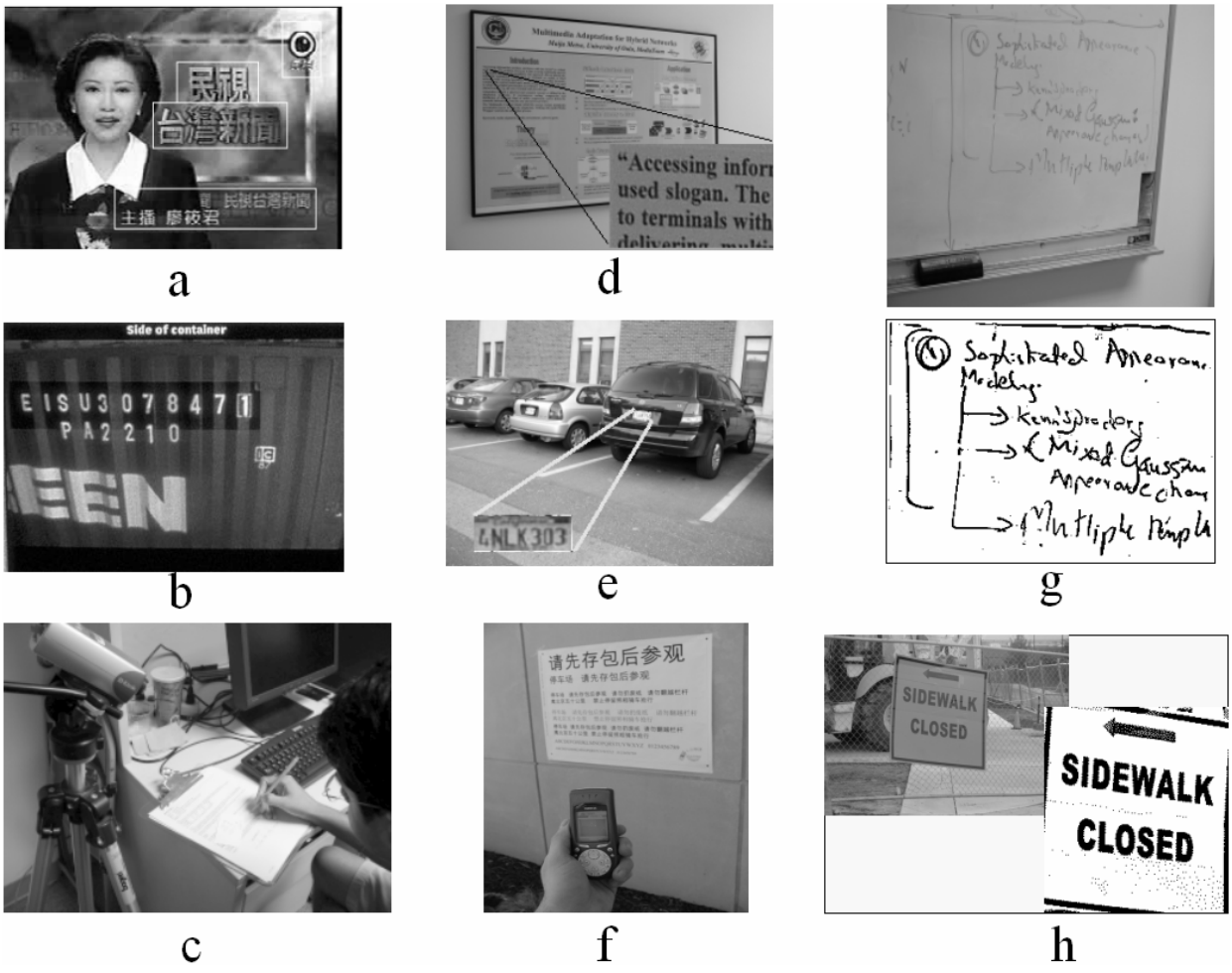

Fig. 4a–g. Sample applications of camera-based document analysis. **a** Video caption text recognition. **b** Cargo container code recognition [49]. **c** Camera-based handwriting recognition [26,70,96]. **d** Poster capturing. **e** License plate reading [8,16,117,119–121]. **f** Sign translation [95,105,106]. **g** Whiteboard reading [88,97]. **h** Road sign recognition

For autofocusing, zooming, and mosaicing, the human–machine interface design is an important part of camera-based document analysis systems. When error is inevitable, user interaction is a must. For mobile devices, a convenient and user-friendly interface is the key to achieving the desired real-time response using very limited hardware resources.

3.3 Image compression

Images and videos can take very large amounts of storage space. Therefore, compression is needed when image-level archiving is necessary. However, current compression schemes are not optimized for document or textual content, nor do they take into consideration that only the textual content is of most interest.

Zunino and Rovetta [117] designrf a vector quantization (VQ) mechanism for license plate images in ALPR. This method not only compresses images but also gives information as to the location of the plate in images. Testing results show that a VQ-based schema outperforms JPEG algorithms in high compression ratio (>30) cases, while JPEG is better in low ratio scenarios.

4 Processing captured images

When considering how to process captured images, one must once again consider the differences between processing *image and video text* and processing *images of structured documents*. This difference will have an impact on techniques needed to process them. Unlike *images of structured documents*, *image and video text* are typically only a subset of a large number of images or video frames, and text detection may be nontrivial. Overall, the procedures involved with processing document images or images of text will require text detection, localization, extraction, geometrical normalization, enhancement/binarization, and recognition. While text detection is typically the first operation, and recognition the last, the rest of the procedures do not have to follow a rigid order. We will leave text detection to Sect. 5 on multiple frame processing since it is a more challenging problem for video.

4.1 Text localization and extraction

Text localization and extraction is a first step and is most prevalent in the analysis of image frames for the analysis of structured documents.

Depending on the application, text localization in images of text can be targeted at caption text, scene text, or both. Caption text usually has strong contrast with background since the producer wants them to stand out clearly. Also often caption text has fixed font size or fixed position. The challenge associated with caption text is that they are of low resolution limited by the video source. Scene text is harder to localize, in general, since it varies arbitrarily in the image. In some specific applications domain knowledge clues may help to find text. For example, in the task of analyzing pictures of paper documents, the document boundaries impose a strong constraint on text position. Most researchers choose to utilize domain-independent text properties to identify text areas. The published algorithms can be broadly classified as gradient feature based, color segmentation based, and texture analysis based. Table 2 at the end of this survey provides a brief summary of these algorithms, including the type of application they are aimed at, training or testing data they use, and results, if reported.

4.1.1 Gradient-feature-based methods. The family of gradient-feature-based text localization methods assumes that text exhibits a strong edge against background, and therefore those pixels with high gradient values are regarded as good candidates for text regions. The methods usually start with either edge detection or other gradient-based feature computation.

One simple gradient feature is the 1D gradient component in either the x or y direction. Wong and Chen [102,117] start by computing the horizontal gradient of each pixel in an image. They assume that the variance of this gradient value in a segment of a horizontal scan line will be large in text areas and small in background. By thresholding the variance, segments of scan lines are selected as potential text segments. Lee and Kankanhalli [49] take a similar approach where the character regions are grown out of vertical scan line segments. Kuwano et al. [48] make use of both vertical and horizontal scan line segments. They construct binary maps consisting of all horizontal and vertical candidate segments, respectively, and keep only regions that exist in both maps with similar shape, size, and position.

General edge detectors make use of 2D gradient information. Both [33] and [34] find the strength of edges using morphological operators. They calculate the difference between the results of a dilation and erosion operators, while a large difference indicates a potential text pixel.

Cai et al. use Sobel edge detectors [6] in all Y, U, and V channels of a color image to find text that is similar to background in luminance but not in color. The results from three channels were combined to make the final decision. The authors also notice that a low threshold is needed for low-contrast text on simple backgrounds, while a high threshold is needed for high-contrast text on complex backgrounds. Therefore, they propose a locally adaptive threshold computation method for the Sobel detector.

Kim et al. present another method of finding edge pixels for potential text regions [45]. In addition to the gradient value, they consider three other gradient-based local features as well: gradient variance, edge density, and edge density variance. In their underlying application of license plate reading, they assume that license plate regions tend to have large gradient variance, high edge density, and low density variance. The features are fed to a simple neural network to determine the potential text pixels.

Lienhart and Wernicle compute the gradient map of an image and apply a neural network classifier on a sliding small window to identify possible text pixels [58]. They adopt a multiresolution approach to find text of various sizes. It is noticed that false alarms tend to happen in isolated layers while text blocks stick out in multiple ones. Therefore, the authors combine the results of different scales to obtain a text salience map.

Instead of finding edges, Hua et al. use a Susan corner detector [35] to find the stroke ends common in characters. Isolated corners in background can be removed due to their low-density property.

A common procedure of all the above methods is to select strong gradient pixels and then group them into text regions, followed by possible postprocessing as a verification.

4.1.2 Color-based methods. Another category of text localization algorithms is based on color segmentation. The designers of either caption text or scene text usually produce the text in a distinguishable color and brightness so it stands out from the background. Under this assumption, after color segmentation, text can be separated from backgrounds based on different color properties they possess. Like gradient-based algorithms, a process is required to group segmented text pixels and verify the results.

Various color segmentation methods have been proposed for general object segmentation in images, and they can be adapted for text localization. Miene et al. use a fast one-pass color segmentation method [66] where pixels are scanned first in the x direction, then in the y direction. Each pixel is compared to its neighboring pixels, and a decision is made if a new region is present. In [56] Lienhart and Stuber report how a split and merge method can be used to perform the segmentation. In another work [57], Lienhart and Effelsberg notice that the split and merge method is sensitive to noise in extremely noisy images such as video frames since those segmentation algorithms do not explore unique text properties and therefore cannot distinguish structured text objects from random objects. With high additive noise, the algorithm will always oversegment text. Their solution is to first oversegment with a fast region-growing algorithm, followed by a merge process where two regions are combined if the border between them does not coincide with

a roughly perpendicular edge or dominant local orientation detected by the inertia tensor.

Another common approach to color segmentation is color quantization. Color quantization has the advantage of smoothing noise among homogeneous regions. After quantization, text areas are assumed to have the same color index. Kim [44] uses iterative dominant color reduction based on color histograms to separate text.

It is often useful to consider more than one color space when doing quantization. In [50] Li et al. propose a multichannel decomposition approach to text extraction from scene images. The original RGB color image is quantized to 27 color indices by taking only the two highest bits from each color component and further mapping four states into three states. A second quantization result is obtained from a filtered image that can eliminate illumination variance to some extent. The multichannel results were fused together in the connected component analysis to generate candidate text regions. Wang and Kangas also use multigroup decomposition [94] where they combine quantization results from the hue component layer, the low saturation layer, the luminance layer, and the edge map. Their quantization method is an unsupervised clustering algorithm.

4.1.3 Texture-based methods. The third category of approaches for text localization is to view text as a unique texture that exhibits a certain regularity that is distinguishable from backgrounds. Humans can identify text of foreign languages even when they do not understand the languages largely due to this distinct texture.

There are several different ways of defining features that grasp the texture. Zhong et al. assume that text areas contain certain high horizontal and vertical frequencies and detect these frequencies directly from the DCT coefficients of the compressed image [115,116]. Wu et al. use Gaussian derivative filters to extract local texture features and apply K-means clustering to group pixels that have similar filter outputs [103,104].

Artificial neural networks (ANN) have been widely used in texture analysis. Jung et al. trained a multiple layer percepton (MLP) that works directly on small image blocks to assign each pixel a text area likelihood [40]. However, typically features were first computed from the original image and fed to an ANN classifier. This step can alleviate the burden of the ANN classifier as well as grasp the most prominent texture features. Li et al. perform wavelet decomposition on multiscale layers generated from the input grayscale image [54,55]. Feature values are the first to third moments of decomposition output. An ANN classifier is trained to label a small window as text or nontext based on the features residing in the window. The multiscale labeling results are integrated into the final result in the original resolution. Clark and Mirmehdi design five statistical texture measurements for text areas and build an ANN classifier to classify each pixel as text or nontext based on the five statistics [12].

Among the methods introduced above, their reported results vary primarily because of differences in the engineering details of the implementation rather than the general methodology. Limited evaluation results have shown that, overall, the performances are not satisfactory. While texture-based methods are more noise tolerant, have a better self-learning ability, and can output a confidence value, they are often computationally expensive compared with gradient- or color-based schemes. A possible solution is to use gradient- or color-based methods as a preliminary detector, and then texture-based methods can be used as the verification.

Another approach is to use OCR as the verification of the text detection. Ohya et al. present a unique solution of combining character detection and recognition [76]. Possible character strokes are formed by local thresholding. Then a relaxation process matches character templates to candidate strokes. Pixels not belonging to any character are discarded. Through the feedback, text localization is verified by OCR results.

4.2 Geometrical normalization

4.2.1 Perspective distortion. As previously suggested, documents that are not frontal-parallel to the camera's image plane will undergo a perspective distortion. When the recognizer is perspective tolerant, rectification is not needed [69]. However, most current text OCR engines are not perspective tolerant. Therefore, perspective correction is necessary. In practice, when perspective is weak, the rectification can be done with simple approximation [34]. In general, suppose the document itself is on a plane; then the projective transformation from the document plane to the image plane can be modeled by a 3×3 matrix in which eight coefficients are unknown and one is a normalization factor. The removal of perspective can be accomplished once the eight unknowns are found. Four pairs of corresponding points are enough to recover the eight degrees of freedom. Under the assumption that text blocks are rectangles in a 3D world, Jung et al. [40] use a straightforward approach to establish four pairs of correspondences and proceed with rectification. However, this method is very prone to error and therefore could only be used when perspective is weak, which makes it unattractive to work with.

For the purposes of OCR, the requirements can be relaxed. As Myers et al. [71] point out, OCR engines are capable of handling different x-to-y scales and are not affected by x- or y-translations. Usually OCR engines are also able to handle different font sizes, which makes the z depth irrelevant, too. Therefore, four unknowns can be removed from the problem. Furthermore, the skew (or rotation) can be estimated by traditional page analysis engines. This leaves only three critical parameters: two perspective foreshortenings along with two axes and a shearing. As described in [84], in a man-made environment where many 3D orthogonal lines exist, the estimation of vanishing points provides a way to recover the perspective. In text documents, parallel text lines, column edge lines, and page boundary lines provide such orthogonal lines. Therefore, the estimation of two van-

ishing points (horizontal and vertical) is enough; these estimations can be relatively robust.

In their study of removing perspective distortion [71, 84], Myers et al. assume that cameras are placed such that vertical edges in scenes are still vertical and parallel in images. The vertical vanishing point where vertical edges intersect is therefore at the infinity of the image plane, while the horizontal vanishing point is in the image plane. They proceed by rotating each text line and observing the horizontal projection profile to find the top and baseline and observing the vertical projection profile to find the dominant vertical edge direction. From the three lines, foreshortening along the horizontal axis and shearing along the vertical axis are determined so that the original text line image is restored. In their work, text lines are restored independently without determining the horizontal vanishing point of the entire text block.

Pilu discusses his method of vanishing point detection in [78]. It closely resembles the bottom-up approach in page segmentation in that it tries to establish connections between nearby connected components. Lines are fit to horizontal linear clues represented by dominant connections. Foreshortening in the x direction is restored once the horizontal vanishing point is found as the intersection of the majority of lines. Vertical linear clues and vanishing point are found in similar ways. However, since horizontal text lines are more abundant than vertical text column boundaries, their approach had some difficulty in dealing with vertical vanishing points.

Clark and Mirmehdi introduce an approximate rectification technique without using vanishing points [11], providing an effective and efficient approximation for OCR purposes. Later [12–14], they propose several methods to estimate the vanishing points given a block of text lines. The horizontal vanishing point is determined through a projection-profile-based voting that is similar to the one used in skew estimation. In the case of fully justified text that presents both linear left and right margins, the intersection of these two margin lines gives the vertical vanishing point. Otherwise, if only a linear left margin is available, the line spacing change caused by perspective is exploited to fix the vertical vanishing point along the margin line. Comparably, the two-margin result is better.

Dance [16] has similar work in estimating vanishing points. His unique contribution is the introduction of probability models such that the vertical vanishing point estimation works with multiple text columns and unknown justification. This method has the potential for graceful degradation in the presence of noise, page curvature, and clutter.

4.2.2 Warping. In some cases, text will appear on curved surfaces. Even with a flatbed scanner, it is not always possible to push a thick book all the way down to get a close contact of every portion of the page with the scanning surface. Kanungo et al. [43] discuss the degradation around the book spine using a cylinder model. In the case of cameras, document pages can take the form of more arbitrary surfaces. As a result of warping, straight lines in the document may appear curved in the image, and squares and rectangles will no longer hold their shapes. This distortion is fundamentally nonlinear and cannot be described by linear transformation as in the case of perspective distortion.

The first category of methods of dealing with images of warped documents aims directly at restoring straight text lines. Zhang and Tan [111] study the case of scanning thick books. To correct the distorted part near the spine, they cluster horizontally nearby connected components to form curved text lines and then vertically move those components that belong to a text line to the same vertical level. The net result is that text lines are straightened. This method, however, will not restore the distorted shape of each individual connected component, so it works when the distortion is not severe. In [112] they extend this work by fitting a quadratic polynomial curve to the distorted text line, which will increase the accuracy of text line straightening. In their test images, the warped portion accounts for about 1/4 of the page. They report that the rectified image can boost the OCR result by more than 10% in both recall and precision.

Approaching the same problem from a different angle, Zhang et al. ([113,114]) propose a method to estimate the cylinder page shape near the spine of an opened book during scanning. In principle, the further away from the scanning surface, the darker the shade. Plus, the warped part of an opened book has a cylinder shape. Therefore, they can use the shading information to induce the shape. The estimated shape not only enables them to restore a flat image but also helps them to remove the shade.

Under the condition that the surface shape can be measured by specialized equipment, it is possible to restore the image of a flat page. In both [5] and [77], the 3D shape of the page is obtained using a structured light method. Brown and Seales [5] propose to model a page by an elastic mesh and flatten the page by pushing the mesh down to a plane. Their method can be applied to arbitrarily bent/stretched/creased paper, which is common in fragile historical manuscripts encountered in their project. Pilu [77] initializes the mesh with the available 3D shape data, then imposes the applicable constraints onto the mesh. After an iterative process, the mesh converges to an applicable state that is closest to the original shape. Using this method, Pilu reports better results than blindly fitting the shape with b-spline surfaces.

Doncescu et al. [21] present similar work using additional lighting equipment. They set up a light grid projector and record the initial positions of the intersection points. The light grid is projected onto the page, and the grid points are detected again. They use the two sets of grid points as control points for a morphing algorithm, which restore the flat image without recovering the 3D shape of the page.

The above range-data-based methods suffer from the requirement of additional specialized equipment. Such equipment may or may not be applicable in other applications (e.g., outdoor). It may also not be desirable when a low-cost, simple, and mobile solution is required. It is

necessary that techniques be developed to do flattening from ordinary images of the warped page, or even better, just from a single image. The answer may lie in the vast literature of shape estimation such as shape from stereo, shape from shade, shape from texture, shape from silhouettes, shape from focus, shape from motion, etc. These techniques should be combined with the specific properties of the document, too. For example, warped pages of opened books often bend on a cylinder parallel to the page allowing for a cylindrical model to be used. Similarly, if we make the assumption that text is laid out horizontally as straight parallel lines on the page, we can use text line features to recover subtle changes in the page structure and unwarp the page back to a plane. For example, a cylinder-model-based approach is presented by Cao et al. in [7], which assumes a frontal view of an open bound book. They estimate the cylinder shape of the page from the text line curves and flatten the page with the model. In their method they do not restrict the warping to be near the book spine or assume specific lighting or shading conditions, which is a step from scanner-based applications toward camera-based applications. However, the requirement of a frontal view and a cylinder shape still needs to be relaxed for it to be really useful.

4.3 Enhancement

The text extracted from either documents or scenes may require enhancement in a number of ways if standard or commercial OCR is to be used. In particular, text should be mapped to binary (black on white), size should be projected to be equivalent to about 12-pt 300-dpi text, edges should be sharpened, and characters should be deblurred when possible. Unlike scanner-based input, where the quality of the image is primarily a function of document quality, the quality of camera text suffers from other external factors (described above) that need to be rectified.

Traditionally, brightness and contrast enhancement are two preliminary tools in image enhancement. Kuo and Ranganath [46] introduce a method for enhancing color and grayscale images and NTSC video frames by contrast stretching. The text in processed images has better visual quality. More advanced enhancement takes into account text properties. Taylor and Dance enhance text image by high-frequency boosting [90] as high frequencies usually correspond to text edges. Chen et al. [10] present a set of Gabor-based filters to measure the text stroke properties and then selectively enhance only those edges most likely to represent text at a specific scale. The net result is that background is smoothed while text edges are preserved so that text detection can be easier. When applying these enhancement techniques, it is important not to create textlike patterns in nontext areas that will cause false alarms in the text detection module. A balance should be carefully managed.

The problem of deblurring is basically concerned with deconvolution, which is ill-posed in its basic form because zeros in the blurring PSF will magnify any noise in input to infinity. Many have worked on solutions to overcome this problem. In both [90] and [75] Tikhonov–Miller regularization is used so that the solution is regularized by a smoothness constraint. Ideally, a smoothness requirement should be weak near character stroke edges and strong in background areas. Instead of adaptively changing the smoothness parameter, [90] and [75] achieved the similar effect by testing the local variance and replacing the pixel with the local average if the local variance is low (i.e., in background). With this method character edges are preserved and the noise in the background is suppressed where it is most noticeable.

In scanner-based OCR systems, image resolution is high, so interpolation is usually unnecessary. A few touching or broken strokes will not significantly affect OCR performance. In low-resolution images, however, character strokes may be only one pixel thick and blended with surrounding backgrounds. Without enhancement a simple binarization will completely remove many strokes. The task of interpolation is typically to increase spatial resolution while maintaining the difference between text and background.

Under the assumption that the Nyquist criterion is met when a continuous object image is sampled by a CCD array, it is theoretically possible to reconstruct the original light field by sinc function interpolation [53]. A perfect high-resolution image can be obtained by resampling at the needed resolution. In practice, where noise is present, perfect reconstruction is meaningless. Bilinear interpolation has been found effective in many instances [42,53,75,79,90].

4.4 Binarization

It has been found that global thresholding is not ideal for camera-captured images due to lighting variation. The alternative is adaptive thresholding, which is a major topic in text image binarization. For example, both [42] and [53] use locally adaptive thresholding to extract text pixels from video frames. Others, including [22] and [39], use adaptive thresholding, too.

A number of adaptive thresholding algorithms originate from Niblack's method. In a well-known survey [92], Trier and Taxt compare 11 locally adaptive thresholding techniques and conclude that Niblack's method is the most effective for general images [92]. In both [90] and [100], Niblack's method is found to be the most effective to extract text. In [90], the comparison showed that Niblack's locally adaptive thresholding with $k = 0$ gave the best result, which is equivalent to thresholding at a local average. In [100], Wolf et al. report that a modified version of Sauvola's postprocessing can effectively suppress the noisy output in pure background areas by the basic Niblack method [100]. In its basic form, Niblack's method has a fixed window size. In [86], Seeger and Dance present an alternate approach to computing locally adaptive thresholds that is in effect a Niblack method with adaptive window size. This is useful when there are different font sizes such as headline title and small text content.

It is also possible to apply binarization without thresholding. Wolf and Doermann [98] obtain the a priori distribution of 4×4 binary cliques in text images from training samples and use a MAP estimator to binarize any 4×4 cliques in input grayscale images [98]. One shortcoming of this approach is that the search space increases exponentially as the clique size is increased in order to capture more micro texture features. To overcome this, [98] uses a simulated annealing technique. However, it requires a lot of representative training samples. In essence, their idea is similar to the joint quantization, interpolation, and binarization method in [25], which does a good job in optimizing the codebook. Fekri et al. apply this VQ-based technique on text images and show better visual results than ordinary interpolation plus thresholding. The VQ-based method faces the same problems as [98], i.e., a large discrete search space when building the vector quantizer and interpolator code book, and the dependence on training samples. Again, simulated annealing is a solution for a good quick approximation. Although more computationally expensive, micro-feature-based binarization shows more potential than traditional adaptive thresholding for domain-specific tasks such as document analysis [101]. As we will see later ([1] in Sect. 5.3), a very similar idea is adopted when it comes to enhancement of multiple images.

4.5 Recognition

Character recognition has long been the fundamental problem in document analysis. However, most systems optimized for scanned documents cannot be applied directly to text acquired with cameras. Therefore, the prevailing research on camera-based document analysis assumes that systems will normalize and enhance text extracted from camera images so that it can be passed directly to commercial OCR and translation software. Only a few reported end-to-end systems are built up in this way (e.g., [57]). However, when much attention is directed to the previous stages of the pipeline, the last recognition stage should not be neglected. A specially designed recognition engine that is perspective tolerant or blur insensitive might give the system the critical boost.

Several researchers have studied the possibility of coupling recognition with previous stages to increase the overall power. For example, Zhang et al. present their OCR engine design, which works directly on graylevel images and reported satisfactory results [110]. Kurakake et al. [47] present an approach to couple character segmentation and OCR to improve both performances. In [47], after text lines are segmented, one character at a time is segmented from left to right, then verified by recognition. If the recognition score is high, the segmentation is accepted. Otherwise, another segmentation position is tried. Similarly, Sato et al. [85] couple character segmentation and recognition. In their approach, all possible cutting positions are found first, and the goodness of a combination of some cuttings is measured by the average character recognition score. A dynamic programming method is used to efficiently find the best segmentation.

In addition to being camera-image unfriendly, available OCR engines are typically developed for workstations and do not fit slim computing devices like PDAs. For OCR to work on resource-limited devices, lightweight algorithms must be handcrafted. They must work on slower CPUs, perhaps with no floating point unit, with less memory, no large storage media for data files, and at real-time speed. Using a set of simple and fast algorithms [79], HP has demonstrated a PDA text recognizer and translator [118].

5 Multiframe processing

In the previous section, we discussed issues related to the processing of a single image known to contain text. Often, however, when processing a sequence of images, there are both new challenges and advantages. The most common case is the well-known video text analysis, but the user may also simply take two or more pictures of the same document, each picture containing either the whole document or part of it. The motivation may be to make sure a clear copy is obtained or a high enough resolution obtained. Many of the same problems are shared by all of these cases including frame selection, text tracking, and multiframe enhancement.

5.1 Frame selection

As the first step of utilizing multiple images in document analysis, images that do not contain text should be eliminated. This step not only saves computation cost in downstream stages, but also reduces the number of false alarms in the text localization step (often text localizers are designed with the goal in mind of not missing any text).

A simple solution is to treat every frame as a potential text frame and apply text region detection to them [48,57,58]. It has the shortcoming of wasting computing power and increasing false alarms. As a simple modification, Kim [44] selects frames at fixed intervals under the assumption that text must appear for a certain amount of time to be visible to the human eye.

In some applications text frame detection is based on shot detection, which basically detects any sudden changes in the visual content, including the appearance and disappearance of text in video. Gargi et al. [29] propose a text frame detection scheme based on the increase in the number of intracoded blocks in MPEG P- and B-frames. Similarly, Kurakake et al. [47] propose a detection method based on the difference of intensity histograms of successive frames (maybe several frames ahead or behind). The position where the histogram changes abruptly is assumed to be a candidate text frame. These methods do not rely on text region detectors. However, they have problems when caption text fades in or out since those situations are easily missed by most shot detectors. In practice, in order to deal with

fade in/out and not to miss any small text, a text region detector should still be used at a certain interval. This method has problems when caption text fades in or out since those situations are easily missed with most shot detectors.

In [109], the goal is to detect score information appearing in sports videos, while ignoring other text such as text in commercials. Zhang et al. found that team names, scores, and other related words usually appear in their fixed positions on the screen. So they adopt a model-based method. Initially, a short sequence of video containing score information is used to train a model about the position and appearance of text. This is a way of detecting the text frames that are of interest, with the slight burden of constructing the model beforehand and a restriction on the types of text frames that can be detected.

5.2 Text tracking

To make full use of the temporal redundancy in video sequences in document analysis, an important idea is to improve the OCR performance by using all instances of the same text in different frames. This requires the tracking of the text over time. For example, in [100], Wolf et al. perform text tracking by simply checking the overlap of detected text blocks between consecutive frames.

Li and Doermann [53] study the tracking of moving text in videos. Once one text object is found in a given frame (called the reference object), an SSD-based (sum of square difference) matching is used to find the best matching block in the next frame. The trace of the text objects is used to rule out false matches that result in random movement. Text boxes are enlarged by a factor of 2 using bilinear interpolation and then matched based on SSD to get subpixel matching precision. Finally, the matched images are averaged and binarized to extract text. To avoid losing the tracking, a recalibration is performed at a fixed interval. Edges in text objects are grouped to get a tighter bounding box of the text, and the new text object is used as a new reference in the tracking process.

Lienhart and Wernicle [58] use a different method to track text objects. A projection-profile-based signature is defined for each text box. In the vicinity of an original text box in a new frame, a best matching box is found based on signature distance. Similar to [53] and [54], a text detector is invoked every five frames to calibrate the tracking. Tracking is continued in the case of a few frame dropouts. Before enhancement, all text boxes are rescaled to a fixed height of 100 pixels. For low-resolution video where text is small, text boxes are interpolated; for HDTV frames, text is downsampled. The rescaled text boxes are aligned at subpixel precision by a SSD-based matching of pixels that have colors close to dominant text color.

5.3 Multiframe enhancement

One advantage of processing multiframe sequences is the ability to integrate over time to improve recognition. When multiple frames are available, the temporal redundancy is often explored to enhance the text quality through a so-called "super-resolution" method. The basic idea of super-resolution algorithms is to construct a clean, high-resolution image from the multiple observation instances, under the fundamental assumption that the constructed super-resolution image should be able to generate the original low-resolution images when appropriately smoothed, warped, and downsampled. In [37], Irani and Peleg demonstrate the effect of super resolution in improving image quality. A process modifies the initial high-resolution guess by iterative back projection (IBP). The success of the method is determined by how accurately the above image formation process can be modeled. Later on, Capel and Zisserman [8] show that Irani's method is superior to the baseline unconstrained maximum likelihood (ML) estimator in noisy cases. They also propose two maximum a priori (MAP) estimators by imposing a priori text property constraints. Li and Doermann [53] utilize a projection-onto-convex-sets (POCS)-based method to deblur scene text to improve readability. The POCS-based method can offer the flexibility of space-varying processing and simultaneously account for blurring due to motion and sensor noise. However, in [53] only linear-space-invariant (LSI) blurring is considered. The extension to more complex cases is not addressed. Elad and Feuer [23] summarize several super-resolution approaches including IBP, ML, MAP, and POCS. They present a hybrid method combining ML and POCS, too. The result on text images shows improvement over original ML and POCS methods.

In a recent study [1], Baker and Kanade analyze the difficulties of the above super-resolution methods and point out that the increase of low-resolution images does not necessarily compensate the decrease of resolution if only smoothness constraints are incorporated into the restoration process. To break the limit, they demonstrate a new algorithm that will learn local texture features from training high-resolution images and enhance these features in low-resolution images. Their work can be viewed as an extension of the MAP approach but has more control over the a priori approaches.

Nearly all super-resolution-based methods require the registration of images to subpixel accuracy, which might be a very challenging task in practice. Brown discusses the general problem of image registration in [4]. He distinguishes three major types of image variations that make registration a problem and establishes the relationship between the variations and techniques that are most appropriately applicable. The three types of variation are variations due to geometry changes (e.g., a change of viewpoint), variations due to optical property changes (e.g., unstable lighting and atmosphere), and variations due to scene changes (e.g., two persons moving in different directions). In a multiframe text image registration scenario, all three types of variations can happen. For example, misalignment can be caused

either by relative movement between the object and the camera, pure optical element adjustment (e.g., zooming), outside lighting change, or image processing in postproduction (e.g., running captions on screen). In some applications of video text enhancement, more constraints can be added so that the registration is manageable. For example, Sato et al. [85] assume that graphic text in a TV news program is static, so no spatial transformation (warping, or translation) is necessary. After locating all text blocks in successive frames, they interpolate the text blocks four times and then integrate them by a temporal min/max operation. The min/max operation will pick the brightest (for normal text) or darkest (for reverse text) background pixels along the time axis, since text pixels do not change their brightness. Wolf et al. [100] present a similar scheme by assuming the text is static. The fusion scheme they choose is the average of multiple frames.

Li and Doermann [53] also present a scheme for enhancing moving text. After identifying the reference frame, they use an image matching technique to track the corresponding text blocks in several consecutive frames. The tracked text blocks are registered to subpixel-level accuracy to improve registration accuracy, then averaged to achieve a clean background and higher resolution. Lienhart and Stuber [56] describe their scheme to handle the moving text, which is based on recognition results: for each character that appears in multiple frames, they gather all corresponding recognition results and select the most frequent one.

All of the algorithms addressed above can only enhance the graphic text under the assumption that the graphic text is either static or has a pure translational motion, such as text scrolling on the screen. The main purpose of the enhancement is denoising. The enhancement of scene text, however, is a much more complex problem and rarely addressed due to often unconstrained motion and possible occlusion. Under the assumption that two images are related only by translation and rotation, Irani and Peleg [37] use an iterative approach to gradually match two images. But their work is not directly aimed at video. Gargi et al. [29] present their video text detection system, which aims at more unconstrained text motion. The movements of text blocks are predicted by motion vectors and completed by a least-square-error search. Their reported results show success in occlusion cases. They also take into account that scene text typically exists in a planar surface in the 3D world, and therefore the motion of text, once projected onto the 2D image plane, should satisfy the planar constraint. A sequence of images can be used to segment different planar surfaces in the image, and the text object is then identified. Later, Crandall et al. [17] introduce a tracking method that uses contour-based shape matching to connect the text object in adjacent frames. Their experiment involves both scene text and caption text; and text undergoes rotation and size changes.

6 Evaluation

Many algorithms have been published for camera-based document analysis, and as new algorithms continue to appear, it is important that benchmarks be set up so these algorithms can be evaluated and compared. Over the past two decades, benchmarking by the OCR community has demonstrated significant improvement in OCR accuracy. Several well-known competitions include the DOE/UNLV annual OCR accuracy test [73,74,82, 83], the Census/NIST OCR Systems Conference [30,98], text detection evaluations of TREC Video Track [125] (TRECVID) and by NIST, and the Robust Reading Competition of ICDAR 2003 [60]. The UNLV conferences evaluate primarily commercial OCR systems and measure the effects of document style and quality on results. The metrics used are primarily character based. The Census/NIST OCR conference focuses on isolated character recognition and demonstrates high accuracy for clean data.

TRECVID is a competition organized by NIST to promote video indexing and retrieval technologies, but text frame detection is evaluated as a standalone "feature." TRECVID chooses to generate a large amount of ground truth data, which only indicates the existence of text in a range of frames rather than identifying each text instance. Each year TRECVID provides a different set of video data for participants, ranging from movies from the 1950s or earlier, to contemporary ABC and CNN news. For example, in TRECVID 2003, more than 130 h of video, mainly from TV sources, were provided. Participants were required to run their algorithms and output the range of frames that had text content, and TRECVID scored the results and released the ground truth data prepared by human judges.

In the ICDAR 2003 Robust Reading Competition [60], a small database (about 500 images) was provided for participants for training and testing their text localization algorithms. Some of the images were extracted video frames, while others were still-scene images taken by digital cameras. XML files were included that associated with each image a set of rectangles indicating the text locations and string contents. Three levels of segmentation ground truth were provided: sentence, word, and character. A sample of the ICDAR data is shown in Fig. 5.

There were five participants in the Robust Reading Competition for 2003, and Lucas et al. summarize the results in [60]. They define precision and recall based on the portion of bounding box overlap. Since the database does not include video sequences but only still images, the temporal aspect is not addressed. The five participants were all in the text locating competition; the other competitions on word reading and character reading had no participants. Among the entries of the text locating competition, the precision ranged from 0.55 to 0.1, and the recall ranged from 0.46 to 0.06. Lucas et al. conclude that even the best performing systems are inconsistent, detecting some text while missing other very similar text in the same scene. Also they are sensible to variation in

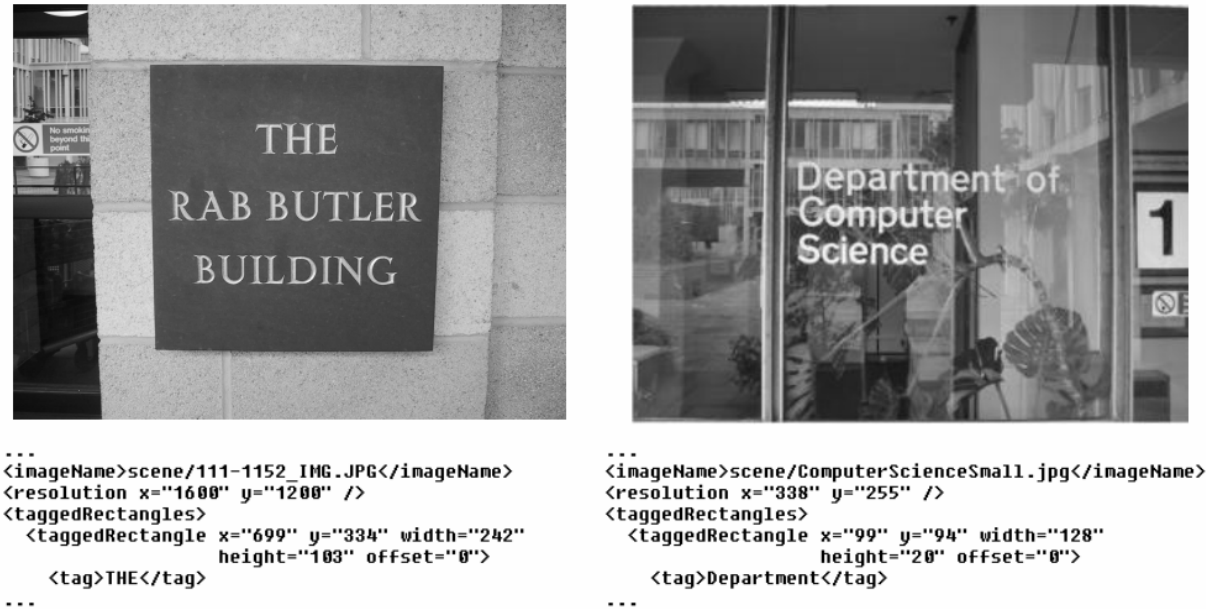

Fig. 5. Sample images and data files from ICDAR 2003 Robust Reading Competition

illumination and scale. Overall, they conclude that text detection is a very challenging problem.

Other evaluation protocols for text detection and localization have been proposed for individual projects, but have not been as widely adopted.

Hua et al. [36] propose to weigh detected bounding boxes against ground truth boxes in terms of predefined detection difficulty, detection importance, and recognition importance. These metrics can be used separately or be combined into an overall quality index. The major problem in this work is that the predefined parameters are subjective and require a considerable amount of manual parameterization.

Mariano et al. [62] believe that no single metric can reveal all aspects of system performance and designed a set of seven metrics to summarize performance. This method requires less ground truth data and is designed for general object detection including faces, people, and vehicles, but it is not tailored specially for text. For example, the legibility problem (if humans can detect text but not read it in a scene), which is considered in Hua's work, is not considered here. This protocol has been tested on detection algorithms from Penn State University and the University of Maryland, respectively. They showed comparable performance in all seven metrics.

In [20], related work on a video performance evaluation resource toolkit (ViPER) is reported. ViPER consists of two parts – ViPER-GT for ground truth generation and ViPER-PE for performance evaluation – and is available as an open source project [124].

Wolf [101] points out that Lucas's scheme used in the ICDAR03 competition did not take into account possible one-to-many correspondences between ground truth objects and experimental results. He also argues that the overlapping area might not always accurately reflect the overlapping quality. He therefore prefers a "crisp" detection number given some detection quality threshold to a possibly ambiguous number that includes overlapping quality. He proposes his own "crisp" version of metrics in his thesis and also combines the ICDAR version and his crisp version as a third metric.

Myers considers the problem of temporal fragmentation and accuracy [64]. Six metrics are designed to measure temporal precision, recall, false alarm, miss rate, fragmentation, and coverage. Spatial information, however, is disregarded. A text object is declared correct if the temporal range is basically correct and if most of the text is correctly OCRed. No evaluation results are reported.

Combining the advantages of all these protocols, there is a general consensus of what a good text localization evaluation protocol should consider, including the following:

Spatial fragmentation. – Some algorithms report all nearby text as a block, while others output separate text lines. Which one is better is merely application dependent. However, fragmentation at the word/character level is usually not ideal.

Bounding box accuracy. – Most algorithms report bounding boxes around text. The boundaries may or may not fall on the exact text edges. The protocol should penalize those cutting through text and those trivial solutions such as a bounding box consisting of the whole image.

Temporal fragmentation and accuracy. – In videos, text exists in a sequence of frames, and there is an issue of tracking in time. Just as with spatial segmentation and accuracy, the protocol should handle imperfect temporal "bounding boxes."

Table 2. Summary of text detection literature showing task type (C for caption text, S for scene text), test data, and results

Ref	Task	Test data	Result	Remarks
Hua 2001	C	90 clips of CNN news, 5 consecutive frames each	244 text boxes: 229 (94%) detected, 18 (7.3%) false alarms	
Kim 1996	C	50 true color 384×288 frames	124 text lines: 107 (86%) detected	
Kurakake 1997	C	100 video clips, 640×480, including running text	All text frames detected; 4.9% false alarms 92% of characters extracted; 82% recognition	Topic frames found without OCR
Kuwano 2000	C	10 h of 8 news programs, 640×480	Of 1,383 caption appearances, 1,314 (95%) detected with 111 (7.8%) false alarms 96% of characters extracted; 76% recognition	Video structure extracted based on caption appearances, without OCR
Lee 1995	S	191 grayscale images, 512×512, of cargo containers	2,096 characters: 1,915 (91%) segmented, 217 (10%) false alarms	Applied to cargo container code reading in ports
Li 1999		45 text blocks from video frames (320×240)	13 (29%) of original blocks, 36 (80%) of blocks enlarged by duplication, and 45 (100%) blocks enlarged by interpolation have OCR output 1,452 characters: 13%, 34%, and 67% recognized, respectively	To show effect of interpolation on OCR
Li 2000	C&S	500 keyframes (320×240) from 22 MPEG video clips, and 75 frames from TV	500 keyframes and 151 text frames: 133 (88%) detected, 81 (38%) false alarms 153 text blocks in 75 TV frames: 142 (93%) detected, 14 (9.0%) false alarms	
Lienhart 1996	C	8 video clips, 384×288, each frame is JPEG	86% of characters contained in candidate text regions in the case of static text on static background	In other cases, where text or background is nonstationary results are in the high 90s
Lienhart 2000	C	22 min videos, in JPEG format, 384×288	Segmentation rate (OCR rate): 96% (76%) on credit sequences, 66%(65%) on commercials, and 99%(41%) on news	
Lienhart 2002	C	23 video clips, total 10 min, 352×240 or 1920×1280, and 7 Web pages stored as images	Text box detection rate improved from 69.5% to 94.7% from tracking	To show effect of text tracking
Messalodi 1999	S	100 grayscale images of book covers normalized to 512×512	91% text line recall, 54% precision	
Miene 2001	C	34 MPEG1 (352×288) video clips of news	91% segmented, 81% recognized	Characters are completely separated from background
Mirmehdi 2001	S	2 sample images, 1440×960 after mosaicing	83-90% recognized (words) in one case	Images were mosaiced from small pieces captured by a 640×480 video camera
Newman 1999	S	6 users working on capturing and OCRing small text segments	6.5% CamWorks OCR error rate at 200dpi (0.6% flatbed with flatbed scanner OCR error rate at 300dpi)	To show feasibility of replacing scanners with cameras
Ortacdag 1998	S	30 CD and book cover color images	25 (100%) text images extracted	

Table 2. (continued)

Ref	Task	Test data	Result	Remarks
Wu 1997	S	48 images from Internet, library, and scanner, including video frames, photographs, newspapers, ads, and checks	Of 21,820 characters (4,406 words), 95%(93%) extracted, 91%(86%) cleaned up; Of extracted, 84% (77%) recognized	
Zhang 2002	C	3 baseball and 1 NBA video clip, all interlaced with commercials	All text bounding areas were correctly learned from initial samples sequences Of 1,134 caption keyframes, 1,130 (99.6%) detected, 22 (1.9%) false alarms 92% OCR rate	Target is scoreboard text; texts in commercials are false alarms Dictionary has 187 words
Zhong 2000	C	2,360 I-frames from 8 MPEG-1 video clips	Of 3,206,936 8×8 compressed blocks: 141,680 labeled as text, 140,983 (99.5%) detected, 50,759 (26.5%) false alarms	

Subjective judgment. – Should a single "$" sign on the paper be counted as text? Or if some scene text is not recognizable by the human eye, should they be counted in text detection? Such questions are not easy to address but are essential.

OCR. – Often the ultimate goal of text detection and recognition is indexing. OCR accuracy is therefore an important component that needs to be considered when appropriate.

Overall, great efforts have been made in evaluating detection techniques. We have ground truth creation tools, large amounts of raw data, and some small ground truth databases. We also have seen several protocols for text detection, localization, and tracking evaluation. The direction of evaluation should be (1) the creation of medium to large size ground truth databases, which is a solid engineering problem; (2) the evaluation of other procedures like geometric normalization, deblurring, and so on; and (3) the objective evaluation of final OCR accuracy or retrieval precision/recall.

7 Summary of grand challenges

Although scanners will not be replaced completely, the ability to capture and process documents with the same ease with which we feed a document into a scanner will revolutionize the way we capture and manage hardcopy documents. We will be able to effortlessly obtain and manage content any time and anywhere, providing for a pervasive environment for hardcopy content. There are a number of key areas that need to be fully addressed, however, before this becomes a reality.

Image quality. – One of the fundamental problems that will need to be addressed, primarily at the device level, is the quality of the images. After years of working with 300-dpi imagery, where lighting has been optimized and device noise is reduced, we are being faced with the challenge of dealing with low-resolution and, in some cases, corrupted data. There is no doubt that cameras could be designed to reduce some of the problem, but in general, document analysis is not a driving force for these devices. Until document analysis is feasible with "standard" off-the-shelf devices that can be purchased at a reasonable cost, the demand will not be sufficient to significantly change the current operation.

Basic algorithms. – As previously stated, camera-based document analysis introduces many new requirements that are not common with scanner-acquired images including dealing with perspective, motion blur, focus, and uneven lighting. A great deal of progress is being made on these problems, but we need to continue to work on them.

Device processing. – Ultimately, our desire for mobility and on-demand processing will require that our document analysis algorithms be ported directly to the imaging devices. Although document capture and offline processing may be the current mode of operation, new applications will drive the demand to, for example, read text and follow up with an online search or database query. This type of real-time processing will require onboard OCR.

New applications. – Perhaps the most interesting thing we can do is to look forward to the new types of applications that will be enabled when we eventually do realize these goals. We will be able to capture, process, and send information from restaurant menus or bus schedules through our cellular phones. We will be able to go into a library and copy, enhance, and ultimately read articles that we need without going to a photocopy machine. We will be able to retrieve and reformat imaged documents automatically to adapt to any device.

Currently our view of document analysis on mobile devices is one of replacing a scanner. Ultimately PDA,

cellular phones, and digital cameras will allow us to work in a way where hardcopy documents are seamlessly integrated into our environment.

References

1. Baker S, Kanade T (2002) Limits on super-resolution and how to break them. IEEE Trans PAMI 24(9):1167–1183
2. Bayer BE () Color image array, US Patent 3971056
3. Bertucci E, Pilu M, Mirmehdi M (2003) Text selection by structured light marking for hand-held cameras. In: In: Proc. ICDAR, pp 555–559
4. Brown LG (1992) A survey of image registration techniques. ACM Comput Surv 24(4):325–376
5. Brown MS, Seales WB (2001) Document restoration using 3D shape: a general deskewing algorithm for arbitrarily warped documents. In: In: Proc. ICCV, pp 367–374
6. Cai M, Song J-Q, Lyu MR (2002) A new approach for video text detection. In: In: Proc. ICIP, pp 117–120
7. Cao H-G, Ding X-Q, Liu C-S (2003) Rectifying the bound document image captured by the camera: a model based approach. In: In: Proc. ICDAR, pp 71–75
8. Chang SL, Chen LS, Chung YC, Chen SW (2004) Automatic license plate recognition. IEEE Trans Intell Transport Syst 5(1):42–53
9. Capel D, Zisserman A (2000) Super-resolution enhancement of text image sequences. In: In: Proc. ICPR, pp 600–605
10. Chen D, Shearer K, Bourlard H (2001) Text enhancement with asymmetric filter for video OCR. In: Proc. ICDAR, pp 192–197
11. Clark P, Mirmehdi M (2000) Location and recovery of text on oriented surfaces. In: Proc. SPIE Document Recognition and Retrieval VII, pp 267–277
12. Clark P, Mirmehdi M (2000) Finding text regions using localised measures. In: Proc. 11th BMVC, pp 675–684
13. Clark P, Mirmehdi M (2001) Estimating the orientation and recovery of text planes in a single image. In: Proc. 12th BMVC, pp 421–430
14. Clark P, Mirmehdi M (2002) On the recovery of oriented documents from single images. In: Proc. Advanced Concepts for Intelligent Vision Systems, pp 190–197
15. Clark P, Mirmehdi M (2002) Recognizing text in real scenes. Int J Doc Anal Recog 4(4):243–257
16. Comelli P, Ferragina P, Granieri MN, Stabile F (1995) Optical recognition of motor vehicle license plates. IEEE Trans Vehicular Technol 44(4):790–799
17. Crandall D, Antani S, Kasturi R (2003) Extraction of special effects caption text events from digital video. Int J Doc Anal Recog 5(2–3):138–157
18. Dance CR (2002) Perspective estimation for document images. In: Proc. SPIE Document Reconition and Retrieval IX, pp 244–254
19. Doermann D (1998) The indexing and retrieval of document images: a survey. Comput Vis Image Understand 70(3):287–298
20. Doermann D, Mihalcik D (2000) Tools and techniques for video performance evaluation. In: Proc. ICPR, pp 167–170
21. Doncescu A, Bouju A, Quillet V (1997) Former books digital processing: image warping. In: Proc. workshop on document image analysis, pp 5–9
22. Du EY, Chang C-I, Thouin PD (2002) Thresholding video images for text detection. In: Proc. 16th ICPR, 3:919–922
23. Elad M, Feuer A (1997) Restoration of a single superresolution image from several blurred, noisy, and undersampled measured images. IEEE Trans Image Process 6(12):1646–1658
24. Etemad K, Doermann DS, Chellappa R (1997) Multiscale segmentation of unstructured document pages using soft decision integration. IEEE Trans Patt Anal Mach Intell 19(1):92–96
25. Fekri F, Mersereau RM, Schafer RW (2000) A generalized interpolative vector quantization method for jointly optimal quantization, interpolation, and binarization of text images. IEEE Trans Image Process 9(7):1272–1281
26. Fink GA, Wienencke M, Sagerer G (2001) Video-based on-line handwriting recognition. In: Proc. ICDAR, pp 226–230
27. Fisher F (2001) Digital camera for document acquisition. In: Proc. symposium on document image understanding technology, pp 75–83
28. Fletcher LA, Kastury R (1988) A robust algorithm for text string separation from mixed text/graphics images. IEEE Trans Pattern Anal Mach Intell 10(6):910–918
29. Gargi U, Crandall D, Antani S, Gandhi T, Keener R, Kasturi R (1999) A system for automatic text detection in video. In: Proc. ICDAR, pp 29–32
30. Geist J, Wilkinson RA, Janet S, Grother PJ, Hammond B, Larsen NW, Klear RM, Burges CJC, Creecy R, Hull JJ, Vogl TP, Wilson CL (1994) The second census optical character recognition systems conference. Technical Report NISTIR 5452, June 1994
31. Gotoh T, Toriu T, Sasaki S, Yoshida M (1988) A flexible vision-based algorithm for a book sorting system. IEEE Trans Pattern Anal Mach Intell 10(3):393–399
32. Haralik RM (1994) Document image understanding: geometric and logical layout. In: Proc. CVPR, pp 385–390
33. Hasan YMY, Karam LJ (2000) Morphological text extraction from images. IEEE Trans Image Process 9(11):1079–1983
34. Hsieh J-W, Yu S-H, Chen Y-S (2002) Morphology-based license plate detection from complex scenes. In: Proc. ICPR, pp 176–179
35. Hua X-S, Chen X-R, Liu W-Y, Zhang H-J (2001) Automatic location of text in video frames. In: Proc. ACM workshop on multimedia: multimedia information retrieval, pp 24–27
36. Hua X-S, Liu W, Zhang H-J (2001) Automatic performance evaluation for video text detection. In: Proc. ICDAR, pp 545–550
37. Irani M, Peleg S (1991) Improving resolution by image registration. CVGIP Graphical Models and Image Processing 53(3):231–239
38. Jain AK, Yu B (1998) Automatic text location in images and video frames. Pattern Recog 31(12):2055–2076
39. Jiang WWC (1995) Thresholding and enhancement of text images for character recognition. In: Proc. IEEE international conference on acoustics, speech, and signal processing, 4:2395–2398

40. Jung K, Kim KI, Han J-H (2002) Text extraction in real scene images on planar planes. In: Proc. ICPR, pp 469–472
41. Jung K, Kim KI, Kurata T, Kourogi M, Han J-H (2002) Text scanner with text detection technology on image sequences. In: Proc. 16th ICPR, 3:473–476
42. Kamada H, Fujimoto K (1999) High-speed, high-accuracy binarization method for recognizing text in images of low spatial resolutions. In: Proc. ICDAR, pp 139–142
43. Kanungo T, Haralick RM, Phillips I (1993) Global and local document degradation models. In: Proc. ICDAR, pp 730–734
44. Kim H-K (1996) Efficient automatic text location method and content-based indexing and structuring of video database. J Vis Commun Image Represent 7(4):336–344
45. Kim S, Kim D, Ryu Y, Kim G (2002) A robust license-plate extraction method under complex image conditions. In: Proc. ICPR, pp 216–219
46. Kuo S-S, Ranganath MV (1995) Real time image enhancement for both text and color photo images. In: Proc. ICIP, 1:159–162
47. Kurakake S, Kuwano H, Odaka K (1997) Recognition and visual feature matching of text region in video for conceptual indexing. In: Proc. SPIE Storage and Retrieval for Image and Video Databases V, San Jose, CA, 3022:368–379
48. Kuwano H, Taniguchi Y, Arai H, Mori M, Kurakake S, Kojima H (2000) Telop-on-demand: video structuring and retrieval based on text recognition. In: Proc. IEEE ICME, New York, pp 759–762
49. Lee C-M, Kankanhalli A (1995) Automatic extraction of characters in complex scene images. Int J Pattern Recog Artif Intell 9(1):67–82
50. Li C, Ding X-Q, Wu Y-S (2001) Automatic text location in natural scene images. In: Proc. ICDAR, pp 1069–1073
51. Li J, Gray RM (1998) Text and picture segmentation by the distribution analysis of wavelet coefficients. In: Proc. ICIP, 3:790–794
52. Li H, Kia O, Doermann D (1999) Text enhancement in digital video. In: Proc. 8th ACM conference on information and knowledge management, pp 122–130
53. Li H, Doermann D (1999) Text enhancement in digital video using multiple frame integration. In: Proc. ACM international multimedia conference, pp 19–22
54. Li H, Doermann D (2000) A video text detection system based on automated training. In: Proc. ICPR, pp 223–226
55. Li H, Doermann D, Kia O (2000) Automatic text detection and tracking in digital video. IEEE Trans Image Process 9(1):147–167
56. Lienhart R, Stuber F (1996) Automatic text recognition in digital videos. In: Proc. SPIE Image and Video Processing IV, 2666:180–188
57. Lienhart R, Effelsberg W (2000) Automatic text segmentation and text recognition for video indexing. ACM Multimedia Syst 8:69–81
58. Lienhart R, Wernicle A (2002) Localizing and segmenting text in images and videos. IEEE Trans Circuits Syst Video Technol 12(4):256–268
59. Lopresti D, Zhou J-Y (2000) Locating and recognizing text in WWW images. Inf Retrieval 2:177–206
60. Lucas SM, Panaretos A, Sosa L, Tang A, Wong S, Young R (2003) ICDAR 2003 robust reading competition. In: Proc. ICDAR, pp 682–687
61. Mao S, Kanungo T (2001) Empirical performance evaluation methodology and its application to page segmentation algorithms. IEEE Trans Pattern Anal Mach Intell 23(3):242–256
62. Margner VF, Karcher P, Pawlowski A-K (1997) On benchmarking of document analysis systems. In: Proc. ICDAR, pp 331–336
63. Mariano VY, Min J, Park J-H, Kasturi R, Mihalcik D, Li H, Doermann D, Drayer T (2002) Performance evaluation of object detection algorithm. In: Proc. ICPR, pp 965–969
64. Myers GK (2003) Metrics for evaluating the performance of video text recognition systems. In: Proc. symposium on document image understanding technology, pp 259–263
65. Messalodi S, Modena CM (1999) Automatic identification and skew estimation of text lines in real scene images. Pattern Recog 32(5):791–810
66. Miene A, Hermes Th, Ioannidis G (2001) Extracting textual inserts from digital videos. In: Proc. ICDAR, pp 1079–1083
67. Mirmehdi M, Palmer PL, Kittler J (1997) Towards optimal zoom for automatic target recognition. In: Proc. 10th Scandinavian conference on image analysis, 1:447–453
68. Mirmehdi M, Clark P, Lam J (2001) Extracting low resolution text with an active camera for OCR. In: Proc. IX Spanish symposium on pattern recognition and image processing, pp 43–48
69. Moravec KLC (2002) A grayscale reader for camera images of XEROX dataglyphs. In: Proc. 13th BMVC, pp 698–707
70. Munich ME, Perona P (2002) Visual input for pen-based computers. IEEE Trans Pattern Anal Mach Intell 24(3):313–328
71. Myers GK, Bolles RC, Luong Q-T, Herson JA (2001) Recognition of text in 3-D scenes. In: Proc. symposium on document image understanding technology, pp 85–99
72. Nagy G (2000) Twenty years of document image analysis research in PAMI. IEEE Trans Pattern Anal Mach Intell 22(1):63–84
73. Nartker TA, Rice SV (1994) OCR accuracy: UNLV's second annual test. INFORM 8(1):40–45
74. Nartker TA, Rice SV (1994) OCR accuracy: UNLV's third annual test. INFORM 8(8):30–36
75. Newman W, Dance C, Taylor A, Taylor S, Taylor M, Aldhous T (1999) CamWorks: a video-based tool for efficient capture from paper source documents. In: Proc. international conference on multimedia computing and systems, pp 647–653
76. Ohya J, Shio A, Akamatsu S (1994) Recognizing characters in scene images. IEEE Trans Pattern Anal Mach Intell 16(2):214–220
77. Pilu M (2001) Undoing paper curl distortion using applicable surfaces. In: Proc. CVPR, pp 67–72
78. Pilu M (2001) Extraction of illusory linear clues in perspectively skewed documents. In: Proc. CVPR, pp 363–368
79. Pilu M, Pollard S (2002) A light-weight text image processing method for handheld embedded cameras. In: Proc. BMVC, pp 547–556

80. Pilu M, Isgro F (2002) A fast and reliable planar registration method with applications to document stitching. In: Proc. BMVC, pp 688–697
81. Plamondon R, Srihari S (2000) On-line and off-line handwriting recognition: a comprehensive survey. IEEE Trans Pattern Anal Mach Intell 22(1):63–84
82. Rice SV, Jenkins FR, Nartker TA (1995) The fourth annual test of OCR accuracy. Technical Report 95-04, Information Science Research Institute, University of Nevada, Las Vegas
83. Rice SV, Jenkins FR, Nartker TA (1996) The fifth annual test of OCR accuracy. Technical Report 96-02, Information Science Research Institute, University of Nevada, Las Vegas
84. Rother C (2000) A new approach for vanishing point detection in architectural environments. In: Proc. 11th BMVC, pp 382–391
85. Sato T, Kanade T, Hughes EK, Smith MA (1998) Video OCR for digital news archive. In: Proc. IEEE workshop on content-based access of image and video database, pp 52–60
86. Seeger M, Dance C (2001) Binarising camera images for OCR. In: Proc. ICDAR, pp 54–59
87. Shim J-C, Dorai C, Bolle R (1998) Automatic text extraction from video for content-based annotation and retrieval. In: Proc. ICPR, pp 618–620
88. Smeaton AF, P, Over: (2002) The TREC-2002 video track report. In: Proc. TREC
89. Stafford-Fraser Q, Robinson P (1996) BrightBoard: a video-augmented environment. In: Proc. conference on computer human interface, pp 134–141
90. Suen H-M, Wang J-F (1996) Text string extraction from images of colour-printed documents. IEE Proc Vis Image Signal Process 143(4):210–216
91. Taylor MJ, Dance CR (1998) Enhancement of document images from cameras. In: Proc. SPIE: Document Recognition V, pp 230–241
92. Trier OD, Taxt T (1995) Evaluation of binarization methods for document images. IEEE Trans Pattern Anal Mach Intell 17(3):312–315
93. Vinciarelli A (2002) A Survey on off-line word recognition. Pattern Recogn 35:1433–1446
94. Wang H, Kangas J (2001) Character-like region verification for extracting text in scene images. In: Proc. ICDAR, pp 957–962
95. Watanabe Y, Okada Y, Kim Y-B, Takeda T (1998) Translation camera. In: Proc. 14th ICPR, pp 613–617
96. Wellner P (1993) Interacting with paper on the DigitalDesk. Commun ACM 36(7):87–96
97. Wienecke M, Fink GA, Sagerer G (2003) Towards automatic video-based whiteboard reading. In: Proc. ICDAR, pp 87–91
98. Wilkinson RA, Geist J, Janet S, Grother PJ, Burges CJC, Creecy R, Hammond B, Hull JJ, Larsen NJ, Vogle TP, Wilson CL (1992) The first optical character recognition systems conference. Technical Report NISTIR 4912, August 1992
99. Wolf C, Doermann D (2002) Binarization of low quality text using a markov random field model. In: Proc. ICPR, 3:160–163
100. Wolf C, Jolion J-M, Chassaing F (2002) Text localization, enhancement and binarization in multimedia documents. In: Proc. ICPR, 4:1037–1040
101. Wolf C (2003) Text detection in images taken from video sequences for semantic indexing. PhD thesis, Institut National de Sciences Appliquées de Lyon, France
102. Wong EK, Chen M-Y () A robust algorithm for text extraction in color video. In: Proc. IEEE international conference on multimedia and expo, pp 797–800
103. Wu V, Manmatha R, Riseman EM (1997) Finding text in images. In: Proc. 2nd ACM international conference on digital libraries, pp 3–12
104. Wu V, Manmatha R, Riseman EM (1999) TextFinder: an automatic system to detect and recognize text in images. IEEE Trans Pattern Anal Mach Intell 21(11):1124–1129
105. Yang J, Gao J, Zhang Y, Waibel A (2001) Towards automatic sign translation. In: Proc. Human Language Technology
106. Yang J, Gao J, Zhang Y, Chen X, Waibel A (2001) An automatic sign recognition and translation system. In: Proc. workshop on perceptive user interfaces (PUI'01)
107. Zandifar A, Duraiswami R, Chahine A, Davis L (2002) A video based interface to textual information for the visually impaired. In: Proc. IEEE 4th international conference on multimodal interfaces, pp 325–330
108. Zappala A, Gee A, Taylor M (1999) Document mosaicing. Image Vis Comput 17(8):585–595
109. Zhang D, Rajendran RK, Chang S-F (2002) General and domain-specific techniques for detecting and recognizing superimposed text in video. In: Proc. ICIP, 1:593–596
110. Zhang J, Chen X-L, Hanneman A, Yang J, Waibel A (2002) A robust approach for recognition of text embedded in natural scenes. In: Proc. ICPR, pp 204–207
111. Zhang Z, Tan CL (2001) Restoration of images scanned from thick bound documents. In: Proc. ICIP, pp 1074–1077
112. Zhang Z, Tan CL (2003) Correcting document image warping based on regression of curved text lines. In: Proc. ICDAR, pp 589–593
113. Zhang Z, Tan CL, Fan L (2004) Estimation of 3D shape of warped document surface for image restoration. In: Proc. ICPR
114. Zhang Z, Tan CL, Fan L (2004) Restoration of curved document images through 3D shape modeling. In: Proc. CVPR, pp 10–15
115. Zhong Y, Karu K, Jain AK (1995) Locating text in complex color images. In: Proc. ICDAR, pp 146–149
116. Zhong Y, Zhang H, Jain AK (2000) Automatic caption localization in compressed video. IEEE Trans Pattern Anal Mach Intell 22(4):385–392
117. Zunino R, Rovetta S (2000) Vector quantization for license-plate location and image coding. IEEE Trans Indust Electr 47(1):159–167
118. http://www.hpl.hp.com/news/2002/apr-jun/translator.html
119. http://www.htsol.com/Products/SeeCar.html
120. http://fire.relarn.ru/personal/andrey/cobra/
121. http://www.roadtraffic-technology.com/contractors/detection/perceptics2/
122. http://www.4digitalbooks.com/products.htm
123. http://donswa.home.pipiline.com/nytimes.digitizing.html
124. http://sourceforge.net/projects/viper-toolkit/
125. http://www-nlpir.nist.gov/projects/t01v/
126. http://www.casioprojector.com/yc400_overview.html