

Sistema de reconocimiento de caracteres para la lectura automática de cheques

El volumen de cheques que deben procesar los bancos actualmente conlleva unos costes que crecen cada año. El interés por automatizar el procesamiento de cheques ha llevado al desarrollo del sistema que se presenta en este artículo. Dicho sistema utiliza técnicas de reconocimiento óptico de caracteres basadas en redes neuronales para leer el valor del cheque.

Introducción

Debido al aumento del número de transacciones que no se realizan en metálico, el volumen de cheques que se procesa actualmente es grande. Alrededor de 50.000 millones de cheques se procesan anualmente en los Estados Unidos, lo que constituye el 60% de los pagos que no se realizan en metálico, según el informe del Federal Reserve Bank [1]. A pesar del aumento de otros mecanismos de pago como la tarjeta de crédito o el pago electrónico, que se automatizan con mayor facilidad, los cheques siguen siendo un mecanismo legal para efectuar pagos y no han experimentado la disminución esperada. Por otro lado, el coste derivado del procesamiento de cada cheque es alto por tratarse de un método lento y que requiere mucho esfuerzo humano. El coste social de cada transacción por cheque se ha estimado entre \$2.78 y \$3.09 [9], siendo un coste que crece cada año.

Al digitalizarse la imagen del cheque y leer automáticamente su valor, se puede agilizar el proceso de validación del cheque, especialmente si se emplean nuevas tecnologías de comunicación basadas en Internet [6]. Además también se reducen los costes asociados a la devolución de cheques ya que el

papel quedaría almacenado en el bando de depósito en lugar de haber sido enviado y retornado en camiones blindados.

Actualmente sólo pueden leerse de manera automática los códigos relacionados con el número de cuenta y número de cheque que están escritos en el cheque con tinta magnética (ver Figura 1). En Estados Unidos se utiliza el conjunto de símbolos E-13B, creado en 1958 por la Asociación Americana de Banqueros, y en España y otros países se utiliza el conjunto de símbolos CMC-7 OCR. El diseño utilizado en estos números se ha optimizado para minimizar los errores del lector magnético [3]. El resto de la información está manuscrita en la mayoría de los casos, más aún cuando muchos de los cheques mecanografiados, generados por sistemas automáticos de pago (como los cheques de nóminas) han sido sustituidos por métodos electrónicos de pago.

El presente artículo describe un sistema desarrollado por Massachusetts Institute of Technology con la colaboración de la Universidad Pontificia Comillas, para leer el valor de un cheque. Parte de la tecnología que se presenta está protegida por una patente (U.S. Patent No. 5633954, 1997).



Rafael Palacios

Escuela Técnica Superior de Ingeniería
ICAI, Universidad Pontificia Comillas



Amar Gupta

Sloan School of Management, Massachusetts Institute of Technology

Figura 1. Tipos de letra estándar para lectura automática del número de cuenta y número de cheque

Ejemplo E-13B

⑆052002166⑆ ⑈11954053⑈ 0204

Ejemplo CMC-7

⑈8635995⑈0065⑈ 0100⑈ 0001449427⑈

Descripción del Sistema de Reconocimiento

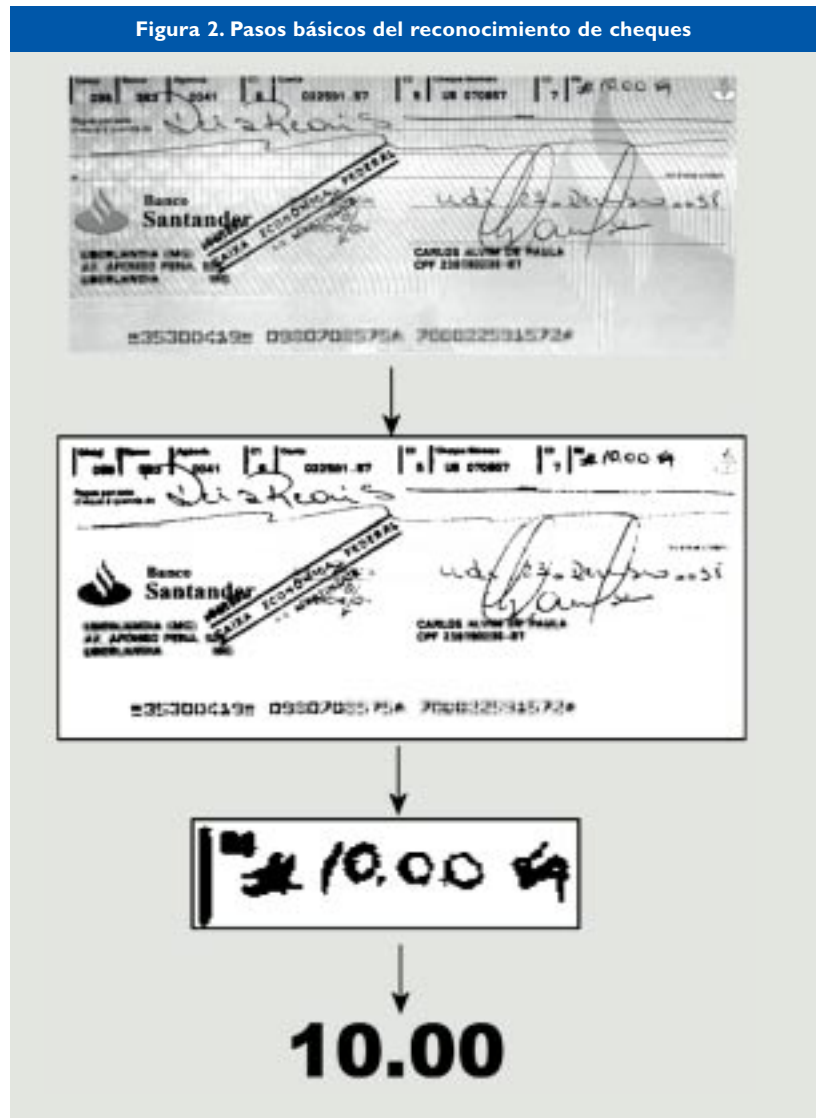
El procedimiento para leer el valor de un cheque comienza escaneando el papel para generar una imagen digital que se almacena en el ordenador. Dicha imagen se obtiene generalmente en escala de grises, ya que la información del color no es relevante en esta aplicación, y se convierte posteriormente a blanco y negro (imagen binaria) para separar el fondo del cheque del texto. Posteriormente el sistema debe localizar la cadena de texto que representa el valor numérico del cheque, lo cual depende del formato del cheque y de pequeños errores de desplazamiento y rotación de la imagen. Después de estos pasos iniciales, el sistema debe realizar la difícil tarea de convertir la imagen del valor del cheque en el número correspondiente (ver Figura 2).

Esta última etapa constituye el paso más difícil, y por lo tanto más interesante, del procesamiento de la imagen del cheque. Nuestra táctica para leer el valor del cheque está esquematizada en la Figura 3 [7].

Una vez localizada la zona de la imagen que contiene el valor del cheque, el sistema debe reconocer los números que allí están escritos. El reto más importante es el proceso de segmentación, que consiste en separar la imagen del valor del cheque en dígitos individuales. En caso de texto manuscrito resulta especialmente crítico separar o agrupar convenientemente los trazos que aparecen en la imagen para formar los dígitos correctos. Por ejemplo, la Figura 4(a) muestra un valor en el cual el número '2' está conectado con el número '9' y debe separarse. Por el contrario en la Figura 4(b) el número '5' está dividido en dos trazos que deben unirse para formar un solo dígito. Nuestro sistema de reconocimiento permite probar distintos algoritmos de separación y de unión hasta encontrar la solución que se reconoce correctamente.

El módulo de reconocimiento de dígitos utiliza una serie de algoritmos de normaliza-

Figura 2. Pasos básicos del reconocimiento de cheques



ción y un clasificador basado en redes neuronales. Este módulo debe ser muy eficiente ya que el bucle general de segmentación lo utiliza intensamente. Finalmente existe un módulo de verificación sintáctica que se encarga de verificar que el resultado del sistema tiene sentido como cantidad monetaria. En este módulo básicamente se comprueba la relación entre número de dígitos y los signos de

Figura 3. Esquema general de la táctica de reconocimiento

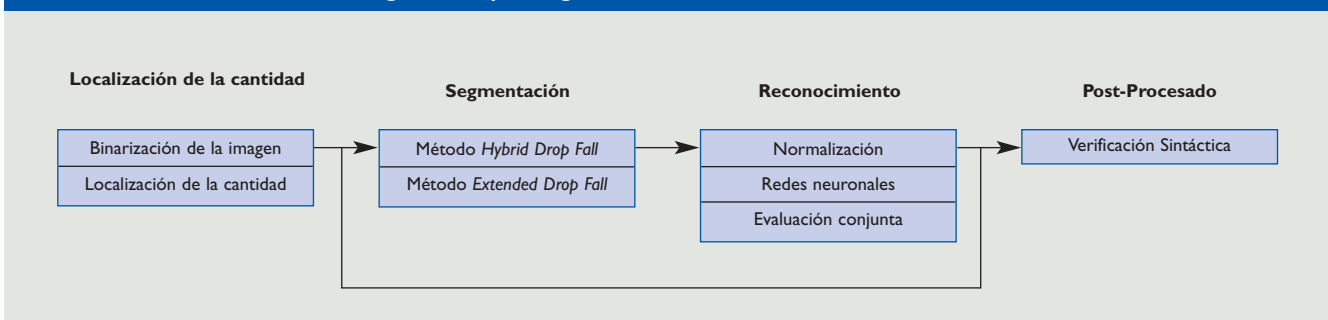


Figura 4. Ejemplos de valores de cheques

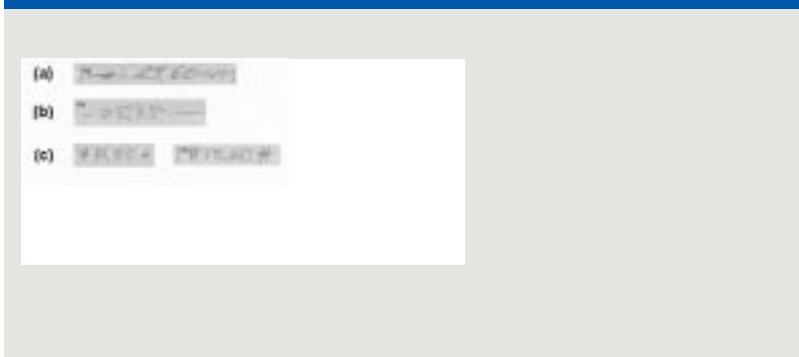
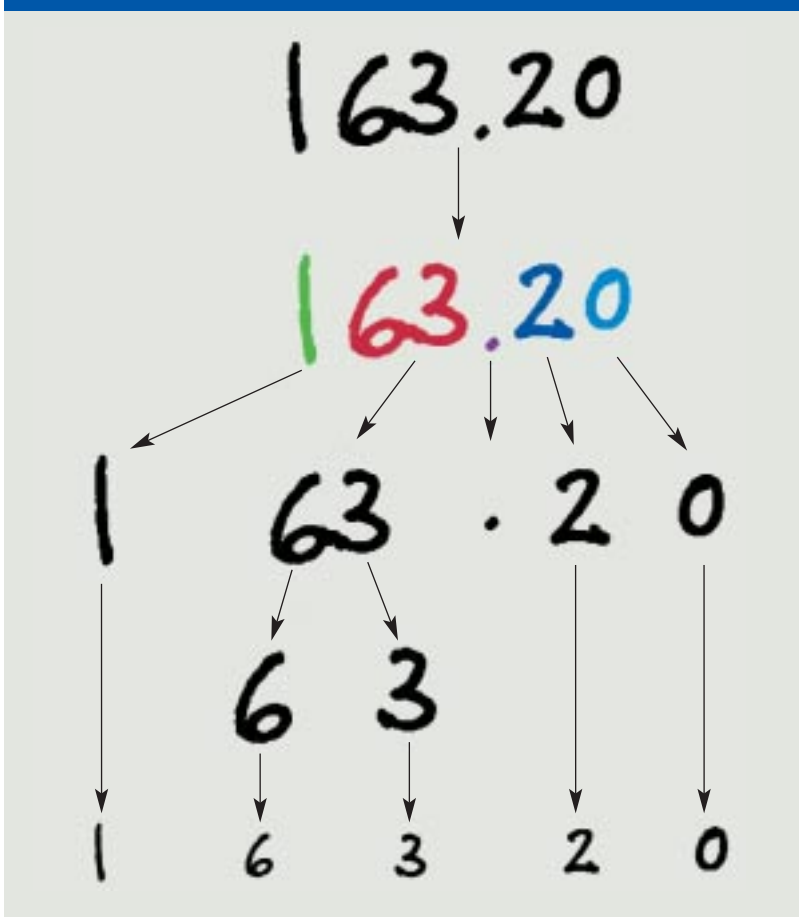


Figura 5. Proceso de segmentación y reconocimiento



puntuación que actúan como separadores decimales o agrupadores. Se describen a continuación el módulo de segmentación y el módulo de reconocimiento, que constituyen las partes fundamentales del sistema.

Segmentación

Una manera bastante efectiva de extraer caracteres de una imagen en blanco y negro es buscar los conjuntos de píxeles que cumplen una relación de conexión entre sí. En la mayoría de los casos este método proporciona caracteres bien segmentados que pueden analizarse sin problemas en el módulo de reconocimiento. Pero si dos o más dígitos se están tocando, lo cual ocurre con bastante frecuencia en el caso de textos manuscritos, entonces resulta necesario separarlos antes de poder reconocerlos (ver Figura 5).

Es difícil separar caracteres que se están tocando ya que no es evidente encontrar el camino óptimo para definir la frontera entre dos caracteres. Varios algoritmos de separación se utilizan en este módulo para poder obtener el camino de separación correcto [7]. Estos algoritmos se basan en la analogía de una gota de ácido que va cayendo por acción de la gravedad y escurre por el borde de los caracteres, pero cuando no se puede desplazar empieza a fundir la tinta y crea un corte de separación. Este algoritmo se puede aplicar en distintas direcciones y con pequeñas variantes, lo que da lugar a diversas maneras de separar los caracteres conectados (ver Figura 6).

El sistema analiza sistemáticamente cada posibilidad de separación hasta encontrar la solución óptima. En principio, si un camino de separación no es válido los segmentos resultantes no serán dígitos y el sistema de reconocimiento los rechaza (por ejemplo en el tercer caso de la Figura 6). Asimismo el sistema de reconocimiento rechaza cualquier símbolo formado por una pareja de dígitos, porque dicho símbolo no coincide con ninguno de los patrones básicos de los dígitos

Figura 6. Resultados de 8 algoritmos Drop-Fall en el caso de un '0' conectado a un '5'



aislados. El hecho de que un símbolo haya sido rechazado aporta una información significativa para el sistema de segmentación, ya que éste decide la estrategia de aplicación de los algoritmos de segmentación en función del éxito de los resultados.

Módulo de Reconocimiento de Dígitos

El módulo de reconocimiento identifica el número que está contenido en cada imagen producida por el método de segmentación. Estas imágenes tienen resolución alta y dimensiones variables, por lo que deben normalizarse antes de llamar al clasificador. El proceso de normalización tiene como objetivo obtener muestras parecidas, en cuanto dimensiones y aspecto, y corregir ciertos efectos de la escritura manuscrita. La normalización incluye corrección de la inclinación, espesor del trazo y tamaño [4]. Se ha comprobado experimentalmente que el proceso de normalización tiene buenos resultados cuando se aplica en 5 fases, tal y como se muestra en la Figura 7.

La corrección de la inclinación convierte en líneas verticales los trazos que se hayan escrito con cierta inclinación, pero sin modificar los trazos horizontales. Esta corrección mejora el rendimiento del clasificador ya que obtiene números más parecidos entre sí para cualquier estilo de escritura. La primera operación de escalado aumenta o disminuye el tamaño de la imagen original para obtener una imagen de tamaño estándar 39x27, que tiene buena resolución. La normalización del espesor se realiza obteniendo el esqueleto y luego dilatándolo, con objeto de conseguir un trazo uniforme de unos 3 pixels de espesor. Finalmente se realiza un segundo escalado para reducir el tamaño a una matriz de 13x9. El resultado final es una imagen pequeña que sólo tiene 117 puntos, pero que representa perfectamente el dígito escrito.

Después del proceso de normalización se aplica un clasificador para interpretar a qué número corresponden los 11 valores bina-

Figure 7. Proceso de normalización

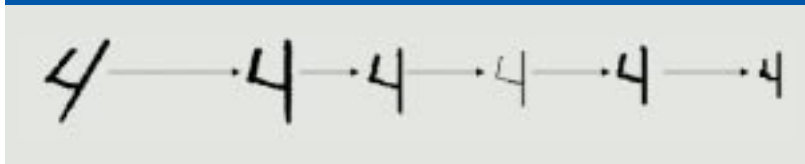
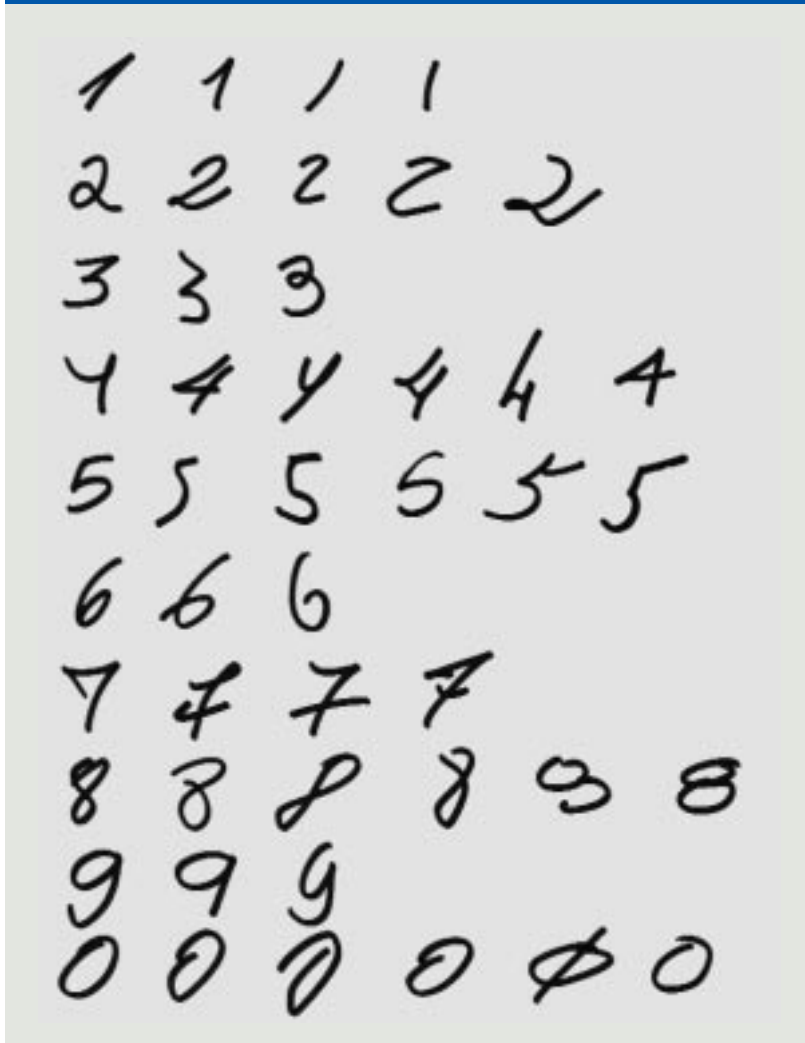


Figura 8. Dígitos obtenidos de cheques de Brasil



rios que se obtienen. En este caso se ha decidido utilizar un clasificador basado en redes neuronales porque generalmente son muy rápidos y ofrecen buenos resultados en el reconocimiento de texto manuscrito. Es importante que el clasificador sea rápido ya que forma parte del bucle principal de segmentación. El número de veces que se utiliza la red neuronal para poder leer el valor de un cheque es variable, ya que bucle de segmentación tendrá que probar diferentes algoritmos de separación de caracteres ligados en función de los resultados obtenidos

por el clasificador. Hay que tener en cuenta que el clasificador debe ser capaz de interpretar una gran variedad de estilos de escritura, especialmente si se tiene en cuenta que existen varias maneras de escribir cada dígito, tal y como se muestra en la Figura 8.

La red neuronal utilizada es un Perceptron Multicapa (MLP), que es una de las redes más empleadas en reconocimiento de caracteres. La estructura es multiconectada y tiene 117 entradas, una capa oculta con 50 neuronas y 10 salidas [5]. Las 10 salidas corresponden a los 10 posibles dígitos, si bien se ha demostrado en otros trabajos que entrenar la red para poder reconocer caracteres ligados y símbolos especiales puede mejorar el nivel de aciertos y sobre todo disminuir la incidencia de reconocimiento erróneo [8]. El nivel de precisión que se obtiene con este tipo de red es muy alto (92.2% de aciertos, con sólo 1.8% de fallos) cuando se entrena con un conjunto de ejemplos grande, tal como la base de datos de caracteres manuscritos de National Institute of Standards and Technology [2]. Sin embargo estos niveles no son tan buenos al reconocer texto manuscrito en cheques o cuando se entrena la red con ejemplos obtenidos en un país y luego se aplica a texto escrito en otro país (como consecuencia de las diferencias en el estilo de escritura de los números) [8]. En cualquier caso el número de lecturas incorrectas, que el parámetro que más preocupa en el sector de las finanzas, disminuye de manera importante al aplicar al aplicar clasificadores paralelos y sistemas de post-procesado [5]. Utilizando varios clasificadores en paralelo (mejor si están basados en distintas técnicas) es posible rechazar un dígito cuando no se obtiene un determinado nivel de acuerdo entre los resultados. Asimismo el post-procesado puede verificar la relación entre grupos de dígitos y los separadores (coma y punto decimal). En general las cifras de valores mayores que mil se agrupan de tres en tres, aunque esto es opcional; por otro lado en la mayoría de los países se admite utilizar dos cifras en la parte decimal. No es complicado establecer unas pocas reglas que permitan rechazar aquellos valores que no cumplan la sintaxis de las cantidades monetarias válidas [5]. No hay que olvidar que rechazar un cheque significa que el sistema no ha podido leerlo automáticamente y por lo tanto una persona debe hacerse cargo del mismo. Esto no es ningún inconveniente ya que la mayoría de los bancos leen manualmente la totalidad de los cheques, sin embargo

es inconveniente garantizar que todos aquellos cheques leídos de manera automáticamente por el sistema sean correctos.

Conclusiones

Este artículo ha presentado un procedimiento de lectura del valor de cheques bancarios especialmente diseñado para reconocer texto manuscrito. Teniendo en cuenta que el volumen de cheques que se utilizan actualmente sigue siendo grande y que se trata de cheques manuscritos en la gran mayoría de los casos, este sistema resulta de gran interés para poder automatizar su procesamiento.

Por otro lado, muchas de las técnicas que se han presentado tiene aplicación inmediata en otros sistemas de reconocimiento de texto manuscrito. Algunos ejemplos son el bucle general de segmentación de caracteres que interactúa con el módulo de reconocimiento, los algoritmos de separación de caracteres ligados, o el procedimiento de normalización de caracteres para producir la información de entrada a la red neuronal. Aunque los detalles técnicos de los algoritmos no se han presentado en este artículo, se han proporcionado las referencias a los artículos originales. ■

Bibliografía

- [1] Federal Reserve Board, "Fed announces results of study of the payments system. First authoritative study in 20 years," Press Release. November 14, 2001.
- [2] M.D. Garris, J.L. Blue, G.T. Candela, P.J. Grother, S.A. Janet, C.L. Wilson. "NIST Form-Based Handprint Recognition System (Release 2.0)", US Dept. of Commerce, Technology Administration. National Institute of Standards and Technology. NISTIR 5959. 1997.
- [3] R. C. González and R. E. Woods, *Digital Image Processing*, 2nd ed., Prentice Hall, Upper Saddle River, NJ, 2002.
- [4] Rafael Palacios, Amar Gupta, and Patrick. S.P. Wang, "Feedback Based Architecture for Reading Courtesy Amounts on Checks", *Journal of Electronic Imaging*, 12(1), pp. 194-202. Jan 2003.
- [5] Rafael Palacios, Anshu Sinha, Amar Gupta "Automatic Processing of Brazilian Bank Checks", submitted to *Machine Vision and Applications*. Oct 2002.
- [6] Rafael Palacios, Amar Gupta, "Web-Based Check Processing", http://www.iit.upco.es/palacios/web_check_processing/. Sep, 2002.
- [7] R. Palacios and A. Gupta, "A system for processing handwritten bank checks automatically", submitted to *Image and Vision Computing*, Feb 2002
- [8] Ashish Singhvi, Rafael Palacios, "Recognizing Handwritten Amounts from Brazilian Checks". Submitted to *Journal of the Brazilian Computer Society*, 2003
- [9] K.E. Wells, "Are Checks Overused?" *Federal Reserve Bank of Minneapolis Quarterly Review* (Fall), pp. 2-12, 1996.