(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 109344941 B
(45)授权公告日 2021.11.30

(21)申请号 201811018687.6
(22)申请日 2018.09.03
(65)同一申请的已公布的文献号
申请公布号 CN 109344941 A
(43)申请公布日 2019.02.15
(73)专利权人 佛山科学技术学院
地址 528000 广东省佛山市南海区狮山镇仙溪水库西路佛山科学技术学院
(72)发明人 马莉
(74)专利代理机构 广州嘉权专利商标事务所有限公司 44205
代理人 王国标
(51)Int.Cl.
G06K 19/06(2006.01)
G06F 8/30(2018.01)

(56)对比文件
CN 105706107 A,2016.06.22
CN 102243714 A,2011.11.16
CN 101908155 A,2010.12.08
JP 2014106887 A,2014.06.09
CN 105706107 A,2016.06.22
唐琳,黄猛,孙明珠.《BMP图像的文本信息隐藏算法》.《电脑编程技巧与维护》.2007,第65-67页.

审查员 林丽香

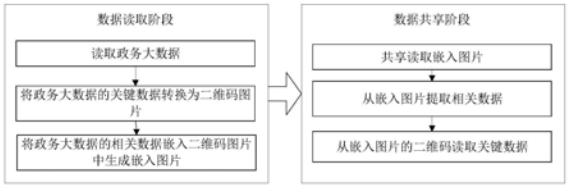
权利要求书2页 说明书10页 附图1页

(54)发明名称

一种大数据安全共享方法及装置

(57)摘要

本发明公开了一种大数据安全共享方法及装置将政务大数据的关键数据,即主键转换为二维码图片,将其他数据嵌入二维码图片中,以共享传递二维码图片的形式安全的进行数据共享,有很强的跨域兼容性,能够方便的进行跨系统、跨平台数据共享,能够实现跨系统、在不同的平台间的数据共享,不需要进行数据加密,保障共享数据的安全性。



CN 109344941 B

1.一种大数据安全共享方法,其特征在于,所述方法包括以下步骤:

数据读取阶段:

步骤1,读取政务大数据;

步骤2,将政务大数据的关键数据转换为二维码图片;

步骤3,将政务大数据的相关数据嵌入二维码图片中生成嵌入图片;

数据共享阶段:

步骤4,共享读取嵌入图片;

步骤5,从嵌入图片提取相关数据;

步骤6,从嵌入图片的二维码读取关键数据;

在步骤1中,所述政务大数据由若干个数据组合组成,每一个数据组合由关键数据和相关数据组成,所述关键数据的值能唯一地标识出政务大数据中的每一个数据组合,所述相关数据为除去关键数据的数据组合其余部分;

在步骤2中,将政务大数据的关键数据转换为二维码图片的步骤为:

步骤2.1,将政务大数据的关键数据按照标准二维码编码规则进行编码;

步骤2.2,将编码生成二维码矩阵;

步骤2.3,将二维码矩阵渲染成二维码图片数据;

在步骤3中,将政务大数据的相关数据嵌入二维码图片中生成嵌入图片包括以下子步骤,

步骤3.1,将政务大数据的相关数据转化为连续的文本比特流;

步骤3.2,将待嵌入的文本比特流嵌入到二维码图片的每个像素的R、G、B分量的冗余位,即R分量的低三位,G分量的最低位,B分量的低二位;

步骤3.2,嵌入文本比特流后生成嵌入图片。

2.根据权利要求1所述的一种大数据安全共享方法,其特征在于,在步骤5中,从嵌入图片提取相关数据的方法为包括以下子步骤,

步骤5.1,从嵌入图片的每个像素的R分量的低三位,G分量的最低位,B分量的低二位组成文本比特流;

步骤5.2,从文本比特流读取政务大数据的相关数据。

3.根据权利要求1所述的一种大数据安全共享方法,其特征在于,在步骤6中,从嵌入图片的二维码读取关键数据的方法源码为,

步骤6.1,构建二维码解码器;

步骤6.2,通过二维码解码器从嵌入图片的二维码读取关键数据。

4.一种大数据安全共享装置,其特征在于,所述装置包括:存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序运行在所述装置的以下单元中:

数据读取单元,用于读取政务大数据;

二维码转换单元,用于将政务大数据的关键数据转换为二维码图片;

数据嵌入单元,用于将政务大数据的相关数据嵌入二维码图片中生成嵌入图片;

数据共享单元,用于共享读取嵌入图片;

数据提取单元,用于从嵌入图片提取相关数据;

二维码读取单元,用于从嵌入图片的二维码读取关键数据;

所述政务大数据由若干个数据组合组成,每一个数据组合由关键数据和相关数据组成,所述关键数据的值能唯一地标识出政务大数据中的每一个数据组合,所述相关数据为除去关键数据的数据组合其余部分;

将政务大数据的关键数据转换为二维码图片的步骤为:

步骤2.1,将政务大数据的关键数据按照标准二维码编码规则进行编码;

步骤2.2,将编码生成二维码矩阵;

步骤2.3,将二维码矩阵渲染成二维码图片数据;

将政务大数据的相关数据嵌入二维码图片中生成嵌入图片包括以下子步骤,

步骤3.1,将政务大数据的相关数据转化为连续的文本比特流;

步骤3.2,将待嵌入的文本比特流嵌入到二维码图片的每个像素的R、G、B分量的冗余位,即R分量的低三位,G分量的最低位,B分量的低二位;

步骤3.2,嵌入文本比特流后生成嵌入图片。

一种大数据安全共享方法及装置

技术领域

[0001] 本公开涉及数据安全技术领域,具体涉及一种大数据安全共享方法及装置。

背景技术

[0002] 在大数据共享传输过程中,很容易导致数据的泄露和被破解,大数据的结构具有多源异构的特点,由不同的重要信息组成,在存储、传输的过程中甚至在共享通信的时候,很容易的被非法用户截获并且轻易读取其中的关键信息。在版权、通信、文件共享等领域时常发生共享信息被非法用户截获等现象,严重危害着数据共享的安全性、在大数据的通信过程中导致了不可控的风险、造成的泄露风险无法预测,在现行的方法中,往往采用对共享数据对称加密等方法解决该问题,但是由于方法固定,加密的数字秘钥很容易被破解。

发明内容

[0003] 为解决上述问题,本公开提供一种大数据安全共享方法及装置,将政务大数据的关键数据,即主键转换为二维码图片,将其他数据嵌入二维码图片中,以共享传递二维码图片的形式安全的进行数据共享。

[0004] 为了实现上述目的,根据本公开的一方面,提供一种大数据安全共享方法,所述方法包括以下步骤:

[0005] 数据读取阶段:

[0006] 步骤1,读取政务大数据;

[0007] 步骤2,将政务大数据的关键数据转换为二维码图片;

[0008] 步骤3,将政务大数据的相关数据嵌入二维码图片中生成嵌入图片,

[0009] 数据共享阶段:

[0010] 步骤4,共享读取嵌入图片;

[0011] 步骤5,从嵌入图片提取相关数据;

[0012] 步骤6,从嵌入图片的二维码读取关键数据。

[0013] 进一步地,在步骤1中,所述政务大数据由若干个数据组合组成,每一个数据组合由关键数据和相关数据组成,所述关键数据的值能唯一地标识出政务大数据中的每一个数据组合,所述相关数据为除去关键数据的数据组合其余部分。

[0014] 进一步地,在步骤2中,将政务大数据的关键数据转换为二维码图片的步骤为:

[0015] 步骤2.1,将政务大数据的关键数据按照标准二维码编码规则进行编码;

[0016] 步骤2.2,将编码生成二维码矩阵;

[0017] 步骤2.3,将二维码矩阵渲染成二维码图片数据。

[0018] 进一步地,在步骤3中,将政务大数据的相关数据嵌入二维码图片中生成嵌入图片包括以下子步骤,

[0019] 步骤3.1,将政务大数据的相关数据转化为连续的文本比特流;

[0020] 步骤3.2,将待嵌入的文本比特流嵌入到二维码图片的每个像素的R、G、B分量的冗

余位,即R分量的低三位,G分量的最低位,B分量的低二位;

[0021] 步骤3.2,嵌入文本比特流后生成嵌入图片。

[0022] 进一步地,在步骤4中,共享读取嵌入图片为在不同地方使用不同计算机、不同软件的用户能够读取到的嵌入图片数据。

[0023] 进一步地,在步骤5中,从嵌入图片提取相关数据的方法为包括以下子步骤,

[0024] 步骤5.1,从嵌入图片的每个像素的R分量的低三位,G分量的最低位,B分量的低二位组成文本比特流,

[0025] 步骤5.2,从文本比特流读取政务大数据的相关数据。

[0026] 进一步地,在步骤6中,从嵌入图片的二维码读取关键数据的方法源码为,

[0027] 步骤6.1,构建二维码解码器;

[0028] 步骤6.2,通过二维码解码器从嵌入图片的二维码读取关键数据。

[0029] 本发明还提供了一种大数据安全共享装置,所述装置包括:存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序运行在所述装置的以下单元中:

[0030] 数据读取单元,用于读取政务大数据;

[0031] 二维码转换单元,用于将政务大数据的关键数据转换为二维码图片;

[0032] 数据嵌入单元,用于将政务大数据的相关数据嵌入二维码图片中生成嵌入图片;

[0033] 数据共享单元,用于共享读取嵌入图片;

[0034] 数据提取单元,用于从嵌入图片提取相关数据;

[0035] 二维码读取单元,用于从嵌入图片的二维码读取关键数据。

[0036] 本公开的有益效果为:本发明提供一种大数据安全共享方法及装置,有很强的跨域兼容性,能够方便的进行跨系统、跨平台数据共享,能够实现跨系统、在不同的平台间的数据共享,不需要进行数据加密,保障共享数据的安全性。

附图说明

[0037] 通过对结合附图所示出的实施方式进行详细说明,本公开的上述以及其他特征将更加明显,本公开附图中相同的参考标号表示相同或相似的元素,显而易见地,下面描述中的附图仅仅是本公开的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图,在附图中:

[0038] 图1所示为一种大数据安全共享方法的流程图;

[0039] 图2所示为一种大数据安全共享装置图。

具体实施方式

[0040] 以下将结合实施例和附图对本公开的构思、具体结构及产生的技术效果进行清楚、完整的描述,以充分地理解本公开的目的、方案和效果。需要说明的是,在不冲突的情况下,本申请中的实施例及实施例中的特征可以相互组合。

[0041] 如图1所示为根据本公开的一种大数据安全共享方法的流程图,下面结合图1来阐述根据本公开的实施方式的一种大数据安全共享方法。

[0042] 本公开提出一种大数据安全共享方法,具体包括以下步骤:

[0043] 数据读取阶段：

[0044] 步骤1，读取政务大数据；

[0045] 步骤2，将政务大数据的关键数据转换为二维码图片；

[0046] 步骤3，将政务大数据的相关数据嵌入二维码图片中生成嵌入图片，

[0047] 数据共享阶段：

[0048] 步骤4，共享读取嵌入图片；

[0049] 步骤5，从嵌入图片提取相关数据；

[0050] 步骤6，从嵌入图片的二维码读取关键数据。

[0051] 进一步地，在步骤1中，所述政务大数据由若干个数据组合组成，每一个数据组合由关键数据和相关数据组成，所述关键数据的值能唯一地标识出政务大数据中的每一个数据组合，所述相关数据为除去关键数据的数据组合其余部分，例如，一个政务大数据的数据组合为企业统一社会信用代码、企业名、联系地址、联系电话，则该政务大数据的数据组合的关键数据为统一社会信用代码，相关数据为企业名、联系地址、联系电话。

[0052] 进一步地，在步骤2中，将政务大数据的关键数据转换为二维码图片的步骤为：

[0053] 步骤2.1，将政务大数据的关键数据按照标准二维码编码规则进行编码；

[0054] 步骤2.2，将编码生成二维码矩阵；

[0055] 步骤2.3，将二维码矩阵渲染成二维码图片数据。

[0056] 进一步地，将编码生成二维码矩阵，将二维码矩阵渲染成二维码图片数据的源码为：

```
//构造二维码写码器
MultiFormatWriter mutiWriter = new com.google.zxing.MultiFormatWriter();
Hashtable hint=new Hashtable();
hint.Add(EncodeHintType.CHARACTER_SET,"UTF-8");
hint.Add(EncodeHintType.ERROR_CORRECTION,com.google.zxing.qrcode.decoder.Err
orCorrectionLevel.H);
//生成二维码
ByteMatrix bm = mutiWriter.encode(txtMsg.Text,
com.google.zxing.BarcodeFormat.QR_CODE, 300, 300,hint);
Bitmap img = bm.ToBitmap();
//要插入到二维码中的图片
Image middlImg = QRMiddleImg.Image;
//获取二维码实际尺寸（去掉二维码两边空白后的实际尺寸）
System.Drawing.Size realSize = mutiWriter.GetEncodeSize(txtMsg.Text,
com.google.zxing.BarcodeFormat.QR_CODE, 300, 300);
//计算插入图片的大小和位置
int middleImgW = Math.Min((int)(realSize.Width / 3.5), middlImg.Width);
int middleImgH = Math.Min((int)(realSize.Height / 3.5),middlImg.Height);
int middleImgL = (img.Width - middleImgW) / 2;
int middleImgT = (img.Height - middleImgH) / 2;
//将 img 转换成 bmp 格式，否则后面无法创建 Graphics 对象
Bitmap bmpimg = new Bitmap(img.Width,
img.Height,System.Drawing.Imaging.PixelFormat.Format32bppArgb);
using (Graphics g = Graphics.FromImage(bmpimg))
{
    g.InterpolationMode =
System.Drawing.Drawing2D.InterpolationMode.HighQualityBicubic;
    g.SmoothingMode = System.Drawing.Drawing2D.SmoothingMode.HighQuality;
    g.CompositingQuality =
System.Drawing.Drawing2D.CompositingQuality.HighQuality;
    g.DrawImage(img, 0, 0);
}。
```

[0057]

[0058]

[0059] 进一步地，在步骤3中，将政务大数据的相关数据嵌入二维码图片中生成嵌入图片包括以下子步骤，

[0060] 步骤3.1，将政务大数据的相关数据转化为连续的文本比特流；

[0061] 步骤3.2,将待嵌入的文本比特流嵌入到二维码图片的每个像素的R、G、B分量的冗余位,即R分量的低三位,G分量的最低位,B分量的低二位;

[0062] 步骤3.2,嵌入文本比特流后生成嵌入图片。

[0063] 优选地,将政务大数据的相关数据嵌入二维码图片中生成嵌入图片的源码如下:

```
// 指向 DIB 的指针
LPBYTE lpDIB;
LPBYTE lpDIBBits;
//读取 DIB 图像
m_hDIB = m_clsDIB.ReadDIBFile(bmpfile);
lpDIB = (LPBYTE) ::GlobalLock((HGLOBAL) m_hDIB);
// 找到 DIB 图像像素起始位置
lpDIBBits = m_clsDIB.FindDIBBits(lpDIB);
// DIB 的宽度和高度
```

[0064]

```
LONG lWidth = m_clsDIB.DIBWidth(lpDIB);
LONG lHeight = m_clsDIB.DIBHeight(lpDIB);
// 计算图像每行的字节数
LONG lLineBytes = WIDTHBYTES(lWidth * 24);
// 从待隐藏的文本文件中获取数据信息
CFile tfile;
tfile.Open(txtfile, CFile::modeReadWrite);
int nFileLen = tfile.GetLength();
unsigned char* lpBuf;
lpBuf = new unsigned char[nFileLen];
```

```
nFileLen = tfile.Read(lpBuf, nFileLen);
m_nFileLen = nFileLen;
tfile.Close();
// 填冗余位时的移位序列
int move1[13] = {6, 5, 2, 0, 7, 4, 2, 1, -2, 6, 4, 3, 0};
// 待隐藏文本文件的字节掩码序列
unsigned char mask1[13]={192, 32, 28, 3, 128, 112, 12, 2, 1, 192, 48, 8, 7};
// 位图文件的字节掩码序列
unsigned char mask2[13]={252, 254, 248, 252, 254, 248, 252, 254, 251, 252, 252,
254, 248};
// 为 1 时 pointer1 指针加一
int add1[13]={0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1};
// 为 1 时 pointer2 指针加一
int add2[13]={1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1};
```

[0065]

```
int pointer1 = 0;
int pointer2 = 0;
int pointer3 = 0;
while(pointer1 <= nFileLen)
{
   // 将文本的比特流嵌入到载体位图
   if (move1[pointer3] > 0)
*(lpDIBBits + pointer2) = (*(lpDIBBits + pointer2) &
mask2[pointer3])|((lpBuf[pointer1] & mask1[pointer3]) >> move1[pointer3]);
   else
*(lpDIBBits + pointer2) = (*(lpDIBBits + pointer2) &
mask2[pointer3])|((lpBuf[pointer1] & mask1[pointer3]) << move1[pointer3] *
(-1));
   if (add1[pointer3] == 1)
      pointer1++;
   if (add2[pointer3] == 1)
       pointer2++;
    pointer3++;
    pointer3 %= 13;
 }。
```

[0066]

[0067] 进一步地，在步骤4中，共享读取嵌入图片为在不同地方使用不同计算机、不同软件的用户能够读取到的嵌入图片数据。

[0068] 进一步地，在步骤5中，从嵌入图片提取相关数据的方法为包括以下子步骤，

[0069] 步骤5.1，从嵌入图片的每个像素的R分量的低三位，G分量的最低位，B分量的低二位组成文本比特流，

[0070] 步骤5.2，从文本比特流读取政务大数据的相关数据。

[0071] 优选地，从嵌入图片提取相关数据的方法的源码如下：

```
// 隐藏的文本信息长度
int nFileLen = m_nFileLen;
LPBYTE lpDIB;
LPBYTE lpDIBBits;
//读取 DIB 图像
m_hDIB = m_clsDIB.ReadDIBFile(bmpfile);
lpDIB = (LPBYTE) ::GlobalLock((HGLOBAL) m_hDIB);
// 找到 DIB 图像像素起始位置
lpDIBBits = m_clsDIB.FindDIBBits(lpDIB);
```

[0072]

```
// DIB 的宽度高度
LONG lWidth = m_clsDIB.DIBWidth(lpDIB);
LONG lHeight = m_clsDIB.DIBHeight(lpDIB);
// 计算图像每行的字节数
LONG lLineBytes = WIDTHBYTES(lWidth * 24);
unsigned char* lpBuf1;
unsigned char* lpBuf2;
lpBuf1 = new unsigned char [lLineBytes * lHeight];
lpBuf2 = new unsigned char [lLineBytes * lHeight];
memset(lpBuf1, 0, lLineBytes * lHeight);
```

```
memset(lpBuf2, 0, lLineBytes * lHeight);
//拼合文件信息字节时的移位序列
int move2[13] = {6, 5, 2, 0, 7, 4, 2, 1, -2, 6, 4, 3, 0};
//位图文件字节的掩码序列
unsigned char mask2[13]={3, 1, 7, 3, 1, 7, 3, 1, 4, 3, 3, 1, 7};
// 为1时pointer1指针加一
int add1[13]={0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1};
// 为1时pointer2指针加一
int add2[13]={1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1};

int pointer1 = 0;
int pointer2 = 0;
int pointer3 = 0;
int pointer4 = 0;
//提取文本信息
while(true)
{
   if(move2[pointer3] > 0)
   lpBuf2[pointer1] |= (*(lpDIBBits + pointer2) & mask2[pointer3]) <<
move2[pointer3];
   else
   lpBuf2[pointer1] |= (*(lpDIBBits + pointer2) & mask2[pointer3]) >>
(move2[pointer3] * (-1));
   if(add1[pointer3] == 1)
   {
      lpBuf1[pointer4] = lpBuf2[pointer1];
   pointer4++;
      // 数据提取完毕，跳出循环
   if(pointer4 > nFileLen)
      break;
```

[0073]

```
                pointer1++;
            }
            if(add2[pointer3] == 1)
            pointer2++;
            pointer3++;
            pointer3 %= 13;
        }
        tfile.Write(lpBuf1, nFileLen);//将提取出的数据写入文件
        ::GlobalUnlock((HGLOBAL) m_hDIB)。
```

[0074]

[0075] 进一步地,在步骤6中,从嵌入图片的二维码读取关键数据的方法源码为,

[0076] 步骤6.1,构建二维码解码器;

[0077] 步骤6.2,通过二维码解码器从嵌入图片的二维码读取关键数据。

[0078] //二维码解码器,从嵌入图片的二维码读取关键数据的方法源码为,

[0079] MultiFormatReader mutiReader=new com.google.zxing.MultiFormatReader();

[0080] Bitmap img=(Bitmap)Bitmap.FromFile(opFilePath);

[0081] if(img==null)return;

[0082] LuminanceSource ls=new RGBLuminanceSource(img,img.Width,img.Height);

[0083] BinaryBitmap bb=new BinaryBitmap(new com.google.zxing.common.HybridBinarizer(ls));

[0084] Hashtable hints=new Hashtable();

[0085] hints.Add(EncodeHintType.CHARACTER_SET,"UTF-8");

[0086] Result r=mutiReader.decode(bb,hints);

[0087] txtmsg2.Text=r.Text。

[0088] 本公开的实施例提供的一种大数据安全共享装置,如图2所示为本公开的一种大数据安全共享装置图,该实施例的一种大数据安全共享装置包括:处理器、存储器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现上述一种大数据安全共享装置实施例中的步骤。

[0089] 所述装置包括:存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序运行在所述装置的以下单元中:

[0090] 数据读取单元,用于读取政务大数据;

[0091] 二维码转换单元,用于将政务大数据的关键数据转换为二维码图片;

[0092] 数据嵌入单元,用于将政务大数据的相关数据嵌入二维码图片中生成嵌入图片;

[0093] 数据共享单元,用于共享读取嵌入图片;

[0094] 数据提取单元,用于从嵌入图片提取相关数据;

[0095] 二维码读取单元,用于从嵌入图片的二维码读取关键数据。

[0096] 所述一种大数据安全共享装置可以运行于桌上型计算机、笔记本、掌上电脑及云

端服务器等计算设备中。所述一种大数据安全共享装置，可运行的装置可包括，但不仅限于，处理器、存储器。本领域技术人员可以理解，所述例子仅仅是一种大数据安全共享装置的示例，并不构成对一种大数据安全共享装置的限定，可以包括比例子更多或更少的部件，或者组合某些部件，或者不同的部件，例如所述一种大数据安全共享装置还可以包括输入输出设备、网络接入设备、总线等。

[0097] 所称处理器可以是中央处理单元(Central Processing Unit，CPU)，还可以是其他通用处理器、数字信号处理器(Digital Signal Processor，DSP)、专用集成电路(Application Specific Integrated Circuit，ASIC)、现成可编程门阵列(Field-Programmable Gate Array，FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等，所述处理器是所述一种大数据安全共享装置运行装置的控制中心，利用各种接口和线路连接整个一种大数据安全共享装置可运行装置的各个部分。

[0098] 所述存储器可用于存储所述计算机程序和/或模块，所述处理器通过运行或执行存储在所述存储器内的计算机程序和/或模块，以及调用存储在存储器内的数据，实现所述一种大数据安全共享装置的各种功能。所述存储器可主要包括存储程序区和存储数据区，其中，存储程序区可存储操作系统、至少一个功能所需的应用程序(比如声音播放功能、图像播放功能等)等；存储数据区可存储根据手机的使用所创建的数据(比如音频数据、电话本等)等。此外，存储器可以包括高速随机存取存储器，还可以包括非易失性存储器，例如硬盘、内存、插接式硬盘，智能存储卡(Smart Media Card，SMC)，安全数字(Secure Digital，SD)卡，闪存卡(Flash Card)、至少一个磁盘存储器件、闪存器件、或其他易失性固态存储器件。

[0099] 尽管本公开的描述已经相当详尽且特别对几个所述实施例进行了描述，但其并非旨在局限于任何这些细节或实施例或任何特殊实施例，而是应当将其视作是通过参考所附权利要求考虑到现有技术为这些权利要求提供广义的可能性解释，从而有效地涵盖本公开的预定范围。此外，上文以发明人可预见的实施例对本公开进行描述，其目的是为了提供有用的描述，而那些目前尚未预见的对本公开的非实质性改动仍可代表本公开的等效改动。

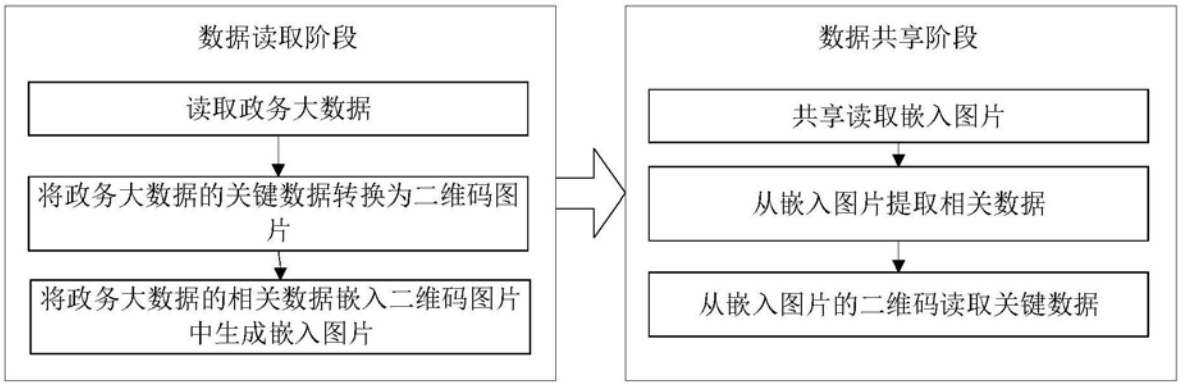

数据读取阶段
读取政务大数据
将政务大数据的关键数据转换为二维码图片
将政务大数据的相关数据嵌入二维码图片中生成嵌入图片
数据共享阶段
共享读取嵌入图片
从嵌入图片提取相关数据
从嵌入图片的二维码读取关键数据

图1

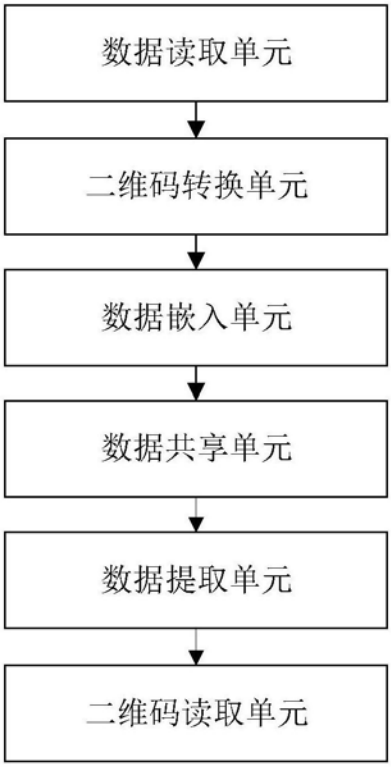

数据读取单元
二维码转换单元
数据嵌入单元
数据共享单元
数据提取单元
二维码读取单元

图2