

# LIS 可视化的标本采集提示实现

## Implementation of LIS Visual Indication of Specimen Collection

何剑虎 (浙江大学 医学院附属妇产科医院 浙江 杭州 310006)

徐松鹤 (浙江省富阳市妇幼保健院 浙江 富阳 311400)

**摘要:** 针对当前临床检验标本采集和核对工作中的识记和质量保证困难,本文介绍了一种在 LIS 标本采集管理子系统中实现可视化的信息提示的方法,其内容包括数据库图像存取、容器图像颜色调整和 Delphi 图像处理,以醒目的彩色图文界面突出了临床检验标本采集环节的工作重点,增强了应用程序的亲合力,提高了软件的易用性和工作效率,有助于减少差错发生。

**关键词:** LIS 标本 可视化 图像 RGB HSV Delphi

### 1 引言

现代医疗诊断,很大程度上依赖于病人体外标本的临床实验室测试。从取得标本到标本送达实验室,检验前阶段是整个检验质量控制中一个容易被忽视却非常重要的环节<sup>[1]</sup>。随着临床检验申请电子化的实施,一些医院的 LIS (Laboratory Information System, 实验室信息系统) 与 HIS (Hospital Information System, 医院信息系统) 实现了一定程度的互联和集成,采集后的各环节可以借助于扫描标本条形码来识别和处理标本,提高了工作效率。但在一般模式下,由于缺少有效的标本采集信息提示,工作质量因人差异较大,人员需要记忆相关知识并有足够的经验才能操作,整个过程除了繁琐的核对之外缺少更有效的质量保证措施,其结果是难以避免处理延误、标本遗漏或混淆等情况。LIS 应用软件实现合适的用户界面和处理将有助于改善和避免这些问题。

本文的可视化的标本采集提示,是根据人对彩色图像更具敏感性的特点,以较为直观的彩色图文形式显示有关信息,来改善 LIS 标本采集管理子系统用户界面,指导工作人员在标本采集环节的操作。它主要应用了数据库图像存取、容器图像颜色调整和 Delphi 图像处理等方法和技术。

### 2 基于 HSV 空间的图像颜色调整

本文所说的容器颜色是指标本容器上用于相互区别的标记颜色,它附着于容器部件,如试管帽。目前许多医院的大部分标本采用具有不同颜色管帽的试管盛

装,易于人眼区别,而管体部分通常为灰色调。在 HSV 颜色空间,如果把容器图像像素颜色的 H 值调整到指定值时,则可以得到想要颜色的图像,图像纹理却不会受损。也就是说,不同颜色的容器图像可以由其它颜色容器经过色调替换处理后获得。因为通常方式下获取的图像是 RGB 颜色空间图像,所以在改变颜色前需要将源图像转换到 HSV 颜色空间。

#### 2.1 RGB 与 HSV 颜色空间<sup>[2]</sup>

RGB 颜色空间值 F,可由红 (R)、绿 (G)、蓝 (B) 三个颜色分量相加混合而获得:

$$F = r[R] + g[G] + b[B] \quad (1)$$

HSV 颜色空间将每种颜色表示为色调 (Hue)、饱和度 (Saturation) 和亮度 (Value) 三个分量。当 V=0, H 和 S 无定义,代表黑色;当 S=0, V=1, H 无定义,代表白色。

#### 2.2 RGB 与 HSV 颜色空间转换<sup>[2]</sup>

RGB 到 HSV 颜色空间的转换公式为:

$$h = \begin{cases} \text{undefined} & \text{if } \max = \min \\ 60^\circ \times \frac{g-b}{\max-\min} + 0^\circ, & \text{if } \max = r \text{ and } g \geq b \\ 60^\circ \times \frac{g-b}{\max-\min} + 360^\circ, & \text{if } \max = r \text{ and } g < b \\ 60^\circ \times \frac{b-r}{\max-\min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max-\min} + 240^\circ, & \text{if } \max = b \end{cases} \quad (2)$$

$$s = \begin{cases} 0 & \text{if } \max = 0 \\ \frac{\max-\min}{\max} = 1 - \frac{\min}{\max}, & \text{otherwise} \end{cases} \quad (3)$$

$$v = \max \quad (4)$$

HSV 到 RGB 颜色空间的转换公式为：

$$h_i = \left\lfloor \frac{h}{60} \right\rfloor \bmod 6 \quad (5)$$

$$f = \frac{h}{60} - h_i$$

$$p = v \times (1 - s)$$

$$q = v \times (1 - f \times s)$$

$$t = v \times (1 - (1 - f) \times s)$$

$$(r, g, b) = \begin{cases} (v, t, p), & \text{if } h_i = 0 \\ (q, v, p), & \text{if } h_i = 1 \\ (p, v, t), & \text{if } h_i = 2 \\ (p, q, v), & \text{if } h_i = 3 \\ (t, p, v), & \text{if } h_i = 4 \\ (v, p, q), & \text{if } h_i = 5 \end{cases} \quad (6)$$

### 3 设计与实现

可视化的标本采集提示内容包括标本种类、标本名称、处理和运送方式等,不同标本种类显示为不同外形和颜色的容器图像。为了减少维护图像数目和信息传输量,以及方便增加或调整,只存储了部分不同形状的容器图像作为素材,而所有容器图像都由软件自动处理素材图像后获得。素材图像在存入数据库之前需要经过裁剪和调整背景、颜色。可视化标本采集提示的实现方法如图 1 所示,包括素材图像存储、素材像获取、容器属性设置、图像颜色调节和界面显示等处理。本文的设计与实现采用了 Delphi7.0 开发工具,数据库管理系统为 Microsoft SQLServer 2000。为达到满意的效果,程序中以直接描绘 TCanvas 对象的方式显示图像和文字。

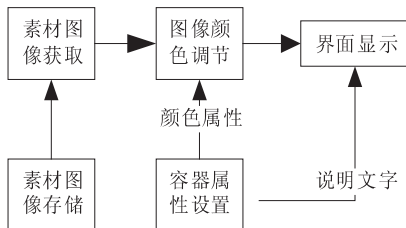


图 1 可视化标本采集提示实现

#### 3.1 容器属性设置

在数据库中建立容器属性表,用来保存不同类型标本容器名称、盛装标本的说明和采集量、容器标记颜色等信息,其形式如表 1 所示(未列出全部字段)。容

器与标本、素材图像之间的关系是:一种容器可用于一种或多种标本,一幅素材图像可生成一种或多种容器的实物图像。表中的颜色值由 RGB() 函数组合了三个颜色分量后获得。

表 1 容器属性表

| 名称  | 颜色值     | 说明  | 颜色转换 | 容器图名   |
|-----|---------|-----|------|--------|
| 紫管  | 8838736 | ... | 是    | Tube   |
| 黄管  | 65535   | ... | 是    | Tube   |
| 尿杯  | 0       | ... | 否    | Cup1   |
| 棉管  | 0       | ... | 否    | Stick1 |
| ... | ...     | ... | ...  | ...    |

软件用户界面如图 2 所示。为方便系统管理员设置,提供颜色选择板和颜色采集器的形式设置颜色属性。

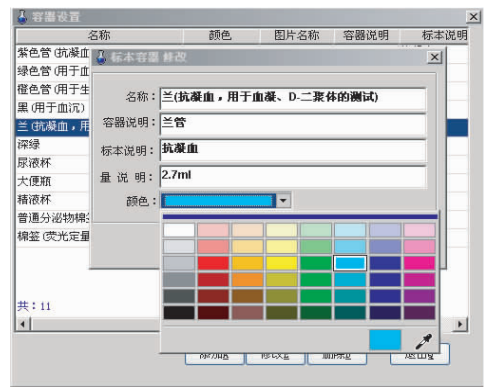


图 2 容器属性设置界面

颜色选择板提供了一些常用的颜色选项。当用户需要选取其它颜色,可点击滴管图标进行全屏颜色采样,其原理是利用 Windows API 像素获取函数 GetPixel() 在鼠标位置采集屏幕像素颜色值。

#### 3.2 素材图像存取

通过数码相机获取的各种不同外形容器源图像,经裁剪和画面调节后作为素材存入数据库。素材图像的存取涉及数据库操作,其方法如图 3 所示。

(1) 素材图像存储:通过 TImage 或 TJPEGImage 对象从文件读入图像数据,然后利用 TADOQuery 对象将 TImage 中的图像作为素材存入 LIS 数据库表的 image 类型(blob 类型)字段。关键代码表示如下:

```

Procedure SaveContainerPic ( Aname: string; APicFile: string );
    
```

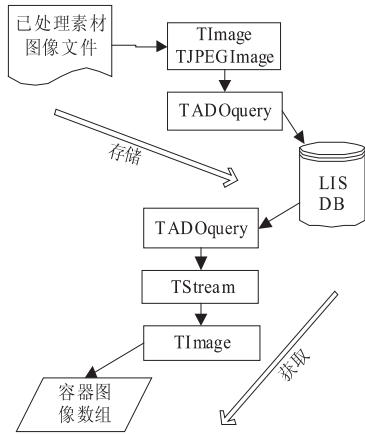


图 3 素材图像存取

```

Var
img: TImage; jpg: TJPEGImage; qPic: TADOQuery;
...
begin
...
If <JPEG 文件> then begin //JPEG 格式文件
    jpg. LoadFromFile ( APicFile ); //从文件读入
    img. picture. bitmap. assign ( jpg ); //转为
bmp 格式
end else //BMP 格式文件
    Img. Picture. LoadFromFile ( APicFile ); //从文件读
入图像
with qPic do
begin
//使用 ADO 查询操作数据库表记录
SQL. Add( `Select PicName, Pic` );
SQL. Add( `From ContainerPictures` );
SQL. Add( `Where...` ); //查询条件
Open;
...
FieldByName( `Pic` ). Assign ( img. Picture. Graph-
ic ); //图像数据存入字段
FieldByName( `PicName` ). AsString := PicName; //
图像名称
Post; //保存
end;
...

```

end;

(2) 素材图像获取: 使用 TADOQuery 类型对象从数据库读取图像表的记录; 通过 CreateBlobStream 函数创建 TStream 对象, 并从 TADOQuery 对象记录中提取图像数据; 使用 LoadFromStream 函数将 TStream 对象中的图像数据导入 TImage 对象, 经过图像尺寸调整后放入一个带 TImage 成员的定义记录数组对象, 供界面显示时使用, 这样可避免频繁访问数据库。关键代码表示如下:

```

procedure loadContainerPic;
var
qry: TADOQuery;
img: TImage; jpg: TJPEGImage; PicName: string;
stm: TStream;
begin
...
with qry do
begin
...;
//从数据库获取所有素材图像及相关说明
Sql. Add( `select * from ContainerPictures` );
Open;
while ( not eof ) do begin
...
PicName := FieldByName( `PicName` )
. asString; //图像名称
stm := CreateBlobStream( FieldByName(
Pic` ), bmRead ); //将图像存放到 stm
img. Picture. Bitmap. LoadFromStream(
stm ); //图像导入 img
ImageResize ( img. picture. bitmap, img,
CONST_CONTAINERWidth,
CONST_CONTAINERHeight ); //调整尺寸
stm. free; //释放 stm
< 将 img 中名为 PicName 的图像放入数组对象 >
next; //下一条记录
end;
end;
end;

```

### 3.3 标本容器图像颜色调节

用户界面需要显示容器时, 先保存图像的数组对

象提取名称匹配的素材图像,然后根据预先设置调节颜色。容器图像的颜色调节处理过程如图 4 所示:首先获取目标颜色的色调 H;再逐个提取源图中像素( $r, g, b$ ),转换到 HSV 空间像素( $h, s, v$ ),判断其色调是否为彩色,如果色调  $h > 0$ ,则令  $h = H$ ,然后将像素( $h, s, v$ )转换到 RGB 空间像素( $r', g', b'$ ),完成所有像素色调检查和变换后显示容器图像。

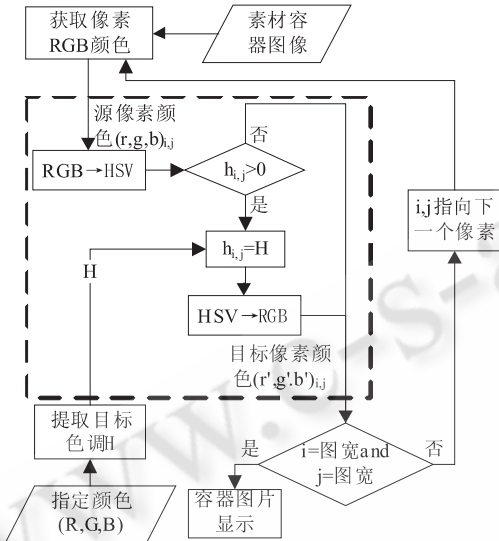


图 4 容器图颜色调整处理

颜色空间的转换函数依据算式(2)~(6)实现,其代码这里不作说明。容器图像颜色调节代码如下:

```

procedure ChangeTubeColor ( bmp: TBitmap; cl:
Tcolor );
//bmp 为素材图像,cl 为目标颜色 RGB 值
var
p: pByteArray; //字节数组
i, j, k: integer;
tH, tS, tV: double; //目标颜色的 HSV 空间值
h, s, v: double; //像素 HSV 空间值
begin
if bmp.Empty then Exit; //无像素,不需处理
RGBtoHSV ( GetRValue ( cl ), GetGValue ( cl ),
getBValue ( cl ), tH, tS, tV ); //指定颜色转换到
HSV 空间
bmp.PixelFormat := pf24bit; //图像格式为 24 位
彩色
//处理图像所有像素点

```

```

for i := 0 to bmp.Height - 1 do begin
p := bmp.ScanLine[i]; //获取一行像素 RGB 颜色数据
for j := 0 to bmp.Width - 1 do begin
k := j * 3; //一个像素颜色由 3 个字节表示,
像素间相对位置差为 3
RGBtoHSV ( p[k + 2], p[k + 1], p[k], h, s,
v );
//若像素为非灰色,即 h > 0,色调改为预设值
if h > 0 then
HSVtoRGB ( tH, s, v, p[k + 2], p[k + 1], p
[k] );
end;
end;
end;

```

### 3.4 界面显示

用户界面采用了 TDrawgrid 和 TImage 可视控件,通过直接描绘其 Canvas 属性对象显示图文信息,即用 TextOut 方法显示文字,用 draw 方法绘图。图像内容输出显示前,调用自定义处理过程 ImageResize,将源图像的尺寸调整为控件显示区域的大小,然后调用 Draw 方法输出显示。

图像尺寸调节只改变图像尺寸,而不改变比例,并使之能合适地显示在指定大小的矩形空间,相关代码如下:

```

procedure ImageResize ( sbmp: Tbitmap { 源位图 }; img:
TImage { 目标位图存放对象 }; w, h: integer );
var
R: TRect;
begin
R.Left := 0; R.Top := 0; //从图布左上端开始画
with sbmp do
if ( Height * w ) > ( Width * h ) then begin //高度方向过长
r.Bottom := h; //纵向占满显示区域
r.Right := trunc ( h * Width / Height ); //宽度按比例调整
end else begin //宽度方向过长
r.Right := w; //横向占满显示区域

```

```

r. Bottom := trunc( w * Height/Width ); //高度按
比例调整
end;
//设定描绘区域大小
img. Picture. Bitmap. Width := r. Right;
img. Picture. Bitmap. Height := r. Bottom ;
//可伸展方式描绘,事实上是按比例缩小或放大
img. Canvas. StretchDraw( r, TGraphic( sbmp ) );
end;

```

图 4 所示是标本采集前的检验医嘱处理程序主界面,其右侧所列的是与执行检验申请有关的标本采集信息提示,包括标本容器的实物图像和有关说明文字。

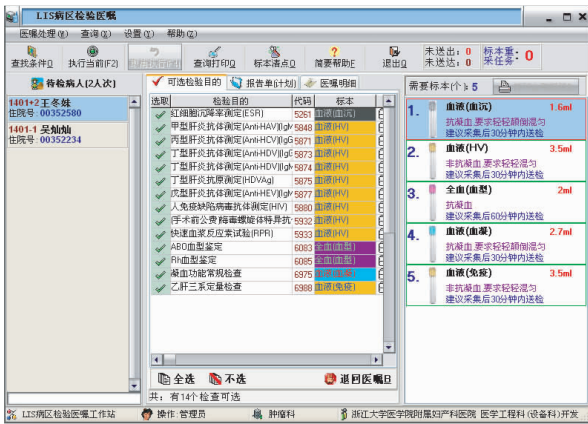


图 5 用户界面

#### 4 结束语

可视化的标本采集提示,以醒目的彩色图文界面突出了临床检验标本采集环节的处理重点,增强了应用程序的亲合力,从而提高了软件的易用性,也方便了标本核对工作,有助于减少差错发生。显示图像均由素材图像动态处理生成,而不需要在系统中维护数量很多的不同容器图像,既可以减少存储空间和传输信息的量,也方便了相关维护工作。系统所需的素材图像是经过专业图像处理软件裁剪、背景和颜色调整的图像文件,虽然素材准备工作不够方便,但是,系统一旦启用后增加图像的情况并不经常需要。

#### 参考文献

- 1 朱美芹. 分析前质量控制的影响因素与流程再造. 中国误诊学杂志,2007,7(18): 4271 - 4272.
- 2 田玉敏,梁若莹. 计算机彩色输入输出设备常用颜色空间及其转换. 计算机工程,2002, 28(9):192 - 200,274.
- 3 刘骏. Delphi 数字图像处理及高级应用. 科学出版社,2003.
- 4 Salton G. Automatic Text Processing. The Transformation, Analysis, and Retrieval of Information by Computer. Addison - Wesley, Reading, MA ,1989.
- 5 Mooney R J, Roy L. Content - based book recommending using learning for text categorization. In: 5th ACM conference on Digital Libraries, pp. ACM Press, New York,2000:195 - 204.
- 6 Chen Z, Fu A W - C, Tong F C - H. Optimal Algorithms for Finding User Access Sessions from Very Large Web Logs. World Wide Web: Internet and Web Information Systems 6,2003. 259 - 279.
- 7 Alata B, Akin E. An efficient genetic algorithm for automated mining of both positive and negative quantitative association rules. Soft Computing - A Fusion of Foundations, Methodologies and Applications 10(3),2005. 230 - 237.
- 8 Antonie M - L, Zaïane O R. Mining Positive and Negative Association Rules: An Approach for Confined Rules. In: Boulicaut J - F, Esposito F, Giannotti F, Pedreschi D, eds. PKDD 2004. LNCS (LNAI) 3202, pp. Springer, Heidelberg ,2004. 27 - 38.

(上接第 11 页)