说明书

手绘场景下的图像识别与智能转化方法、系统及计算机可读介质

技术领域

本发明属于图像识别与人机交互技术领域，涉及手绘流程草图的识别及其计算机可编辑标准格式的生成，具体涉及手绘场景下的图像识别与智能转化方法、系统及计算机可读介质。

背景技术

当前计算机技术及拍摄技术的迅速发展，为绘制图形及人机交互提供了便利条件。但是在日常工作及生活中，人们依然倾向于使用白板、黑板、纸张或便携式电子设备等进行实时手绘流程图，再通过人工识别与操作，将手绘草图转化为可编辑图形。传统的人工转化进行人机交互的方式效率较低，转化时不仅需要用户对于绘图软件本身具有较强的熟悉，包括软件的自身布局，软件中各类形状的名称及位置，软件及其形状的各个属性的功能及调节方式等，还需要人工手动使用鼠标拖动各个形状进行使用，同时还需要记忆大量快捷键进行快速操作，严重降低了用户的绘图效率，使得用户无法花费更多的时间在流程图本身质量上，不能完成满足人机交互的需求，因此探索自然、高效、智能化的人机交互方式已成为计算机研究领域的重要议题之一。

在人机交互领域中，端到端的功能实现是最为便利的人机交互通道，备受关注。用户仅需将原始需求输入，端到端的功能系统即可实时输出用户最终需求范式，如直接输入原始手绘流程图图片，端到端的功能系统实时输出最终指定软件的可编辑图形。用户仅需根据自身额外需求对最终输出进行小范围精确改动。因此，如何实现利用原始手绘流程图图片直接生成可编辑图形的端到端功能系统，

以大幅提高软件绘图人机交互操作的效率，是亟需解决的技术问题。

然而，目前针对手绘流程图转化为软件可编辑图形的研究还十分空白，绝大多数的方法都是只停留在手绘流程图整体识别阶段，且其中一部分方法仅对于手绘流程图中的基本形状进行定位与识别，忽略了各个形状之间的相互连接关系；而另一部分方法虽然考虑了基本形状的定位识别与各个形状之间的连接关系，但是却要求用户使用特定的连接关系画法，因此在使用该类方法时需先了解方法本身所规定的流程图形状、连接线等画法，额外限制了用户的绘图效率。尽管存在部分限制条件较少的流程图定位识别方法，但是存在训练集数量及覆盖范围小，绝大多数公开训练集中的流程图数量仅在 500-800 之间，且大多数流程图中的形状个数不超过 10 个，缺乏对于日常工作生活中常见的中大型流程图的覆盖，以及未能有效与常用绘图软件相结合，对于常见的绘图软件，一方面是由于缺乏轻量实用的开发工具进行集成开发，如 Visio 等，另一方面是由于现存的开发工具门槛较高，并不与流程图识别方法的输出直接兼容，集成需花费的时间和精力较高，如 PowerPoint 等。因此，目前针对手绘流程图转化技术的研究的准确性、效率以及可用选择均较低。

发明内容

为了克服上述现有技术存在的不足，本发明的目的在于提供手绘场景下的图像识别与智能转化方法、系统及计算机可读介质，能够有效解决在包括部门开会、小组讨论、学习笔记等各种手绘场景下的手绘流程图转化为标准可编辑图形过程中存在的多种问题，并通过提供不同版本的网络架构实现转化速度与精度的平衡。尤其是在日常开会记录的工作场景下使人机交互更为便捷，通过拍摄原始流程图片上传，实现真正端到端的手绘图片转化，并集成了外部的 OCR 资源包进行手写字体的识别，用户可以根据自己的特殊需求进行小范围的

精细调整，通过自动转化与小范围人为干预，达到对于各类手绘流程图的快速、高效、准确地转化，解决了现有技术中存在的手绘流程图转化技术的研究的准确性、效率以及可用选择较低的问题。

为了实现上述目的，本发明采用的技术方案是：

手绘场景下的图像识别与智能转化方法，包括以下步骤；

步骤 1：手绘流程草图采集：通过相机实时拍摄，实时扫描当前的手绘流程草图，或直接软件绘制流程图，采集当前时刻的手绘流程图像信息，输入至计算机，实现实时的手绘流程图的采集与传输；

步骤 2：获取步骤 1 得到的手绘流程图像信息，通过定位形状位置和识别形状类别步骤，最终输出各个预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状数据；

步骤 3：获取步骤 2 的预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状数据，通过计算机绘图软件智能展示，输出指定软件的可编辑图形；

步骤 4：OCR 模块搭载，通过预留外部接口，实现与 OCR 功能包的集成进行手绘流程图文字识别，并与软件接口对接，将步骤 3 得到的指定软件的可编辑图形，实现文字自动生成，最终输出完整的可编辑图形。

所述步骤 2 中定位形状位置和识别形状类别，包括形状坐标检测、形状类型识别、箭头特征点定位、箭头指向对象估计、设置置信度阈值；

形状坐标检测：将采集的原始手绘流程图像信息输入至深度级联神经网络模型中进行特征提取，然后进行形状候选框筛选，输出每个形状的位置坐标框；

其中，所述深度级联神经网络模型包括用于提取图像全局特征的 FPN 网络(主要由 CNN 网络组成)和用于生成候选框的 RPN 网络(主要由 CNN 网络组成)；

形状类型识别：将所述每个形状的位置坐标框输入至并联神经网络模型中进行特征提取，然后进行形状分类，结合形状位置坐标框，输出最终位置的形状框；

其中，所述并联神经网络模型包括用于形状坐标框回归、形状类型预测和关键点回归的三并行网络组成，三者都是由全连接网络组成；形状位置坐标框是由(x_1, y_1, x_2, y_2)的四元组组成，表示坐标框的左上角和右下角坐标，而最终位置的形状框是由$(box, class, score)$的三元组组成，其中box表示前述的形状位置坐标框，$class$表示该形状的类型，$score$表示该形状为类型$class$的概率。

箭头特征点定位：将箭头转化为连接形状对象的因果关系，并使用始末特征点表示对应的因果关系，通过约束模型对输出的最终位置的箭头形状框中的特征点进行检测，标注出最终位置的箭头形状框中的箭头二维特征关键点，实现箭头特征点定位；

其中，所述的连接形状对象包括形状坐标检测及形状类型识别所预测的所有形状；因果关系指预测的箭头关键点的对应关系，靠近箭头起始的关键点(始特征点)为因，靠近箭头终点的关键点(末特征点)为果；所述的约束模型主要包括边框限制，即通过比较关键点与对应的箭头坐标框，限制关键点位于箭头坐标框内；

箭头指向对象估计：对于输出的每一个箭头形状框中的特征点与其周边的形状，根据形状位置、关键点位置以及像素坐标系下两者之间存在的几何关系，使用智能算法估计箭头关键点的归属关系；

其中，箭头关键点的归属关系是指关键点对应到具体预测的形状；智能算法估计主要包括形状确定与最短距离：形状确定通过预测的矩形坐标框与预测类型形状的对称性，确定形状的顶点坐标，从而确定边；最短距离通过计算箭头关

键点到各个形状的边的最短距离，确定关键点对应的具体形状；

设置置信度阈值：对比各个预测形状在整体图像上的坐标位置及其置信度，结合箭头指向对象估计的确信度，选取形状位置合理、箭头指向对象明确，置信度高于设定阈值的识别结果作为最终的形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状数据；

所述步骤 3 中计算机绘图软件智能展示，包括深度神经网络与软件接口实现、箭头关键点因果转换、整体轮廓智能排版；

深度神经网络与软件接口实现：结合计算机绘图软件特定文档接口输入要求，提取神经网络原始输出，并将其转化为软件接口所规定的输入形式，编写接口函数自动根据预测形状生成可编辑图形；

其中，深度神经网络的原始输出指前述的并联神经网络的输出，包括形状坐标框、形状类型和箭头关键点；

箭头关键点因果转换：通过箭头指向对象估计，进而推断出对象之间的连接关系，起始点为因，终止点为果，编写函数实现形状对象间的关系连接；

整体轮廓智能排版：对于生成的原始可编辑图形，对比各个形状的绝对与相对位置，采用启发式聚类算法实现横向与纵向对齐；形状大小标准化、一致化完成自动智能排版，同时提供软件内置智能排版算法进行选择；

其中，形状的绝对位置是形状的坐标所表示的位置；形状的相对位置是形状在整体下的方位位置(如左上角)。

所述计算机绘图软件为 PPT 或 Visio。

所述步骤 1 中手绘流程图像信息包括相机拍摄的原始 RGB 图像、经过扫描软件二值化加工的预处理图像，或在电子设备上直接绘制生成的图片经过编程语言(python 的 opencv 工具包)读入的三维矩阵。

所述步骤 2 中形状坐标检测具体为：

将采集的手绘流程图的图像通过形状/背景二分类器输出得到各个可能含有形状的位置坐标框，将得到的各个可能含有形状的位置坐标框输入至深度级联神经网络模型中的第一个子网络结构中进行特征提取，输出众多候选框，对输出的候选框使用边界框回归方法合并重叠的候选框，得到校正后的候选框；校正后的候选框分为两部分，第一部分用来判断当前校正后的候选框内图像是否存在形状，第二部分表示当前校正后的候选框内形状位置相对真实目标的形状框位置的偏移；

将通过第一个子网络得到校正后的候选框作为输入，通过深度级联神经网络模型中的第二个子网络结构，输出粗预测的候选框，使用边界框回归方法合并重叠的粗预测的候选框，得到第二次校正后的候选框；

将通过第二个子网络结构得到的第二次校正后的候选框作为输入，通过深度级联神经网络模型中的第三个子网络结构，输出最终位置的形状位置坐标框；

所述步骤 2 中形状类型识别具体为：

通过第三个子网络结构得到的形状位置坐标框输入至深度神经网络模型中进行特征提取，通过形状多分类器输出得到该形状位置框内为各个形状的概率，将得到的各个形状所属的概率通过 softmax 层进行归一化，计算如下式所示：

$$p_i = \frac{e^{z_i}}{\sum_{j}^{N} e^{z_j}}$$

式中：z_i、z_j 为分类器原始输出，p_i 为归一化概率，N 为形状类别总数。

所述形状/背景二分类器的交叉熵损失函数 L_i^{truth}，表达式如下：

$$L_i^{truth} = -(y_i^{truth}\log(p_i) + (1 - y_i^{truth})(1 - \log(p_i)))$$

$$y_i^{truth} \in \{0,1\}$$

式中：y_i^{truth} 为形状/背景的实际标签，p_i 为形状的概率；

所述边界框回归方法使用非极大值抑制合并重叠的候选框，得到校正后的候选框，边界框回归方法通过计算候选框的背景坐标 $\hat{y}_i^{box}$ 与实际的背景坐标 y_i^{box} 之间的欧式距离，计算回归损失函数 L_i^{box}，表达式如下：

$$L_i^{box} = \| \hat{y}_i^{box} - y_i^{box} \|_2^2$$

$$y_i^{box} \in R^4$$

式中：$\| \|_2^2$ 表示欧氏距离的 L2 损失函数，y_i^{box} 表示第 i 个真实形状框的位置坐标，$\hat{y}_i^{box}$ 表示对应的第 i 个预测形状框的位置坐标，R^4 表示四维实数空间；

所述形状多分类器的交叉熵损失函数 L_i^{class}，表达式如下：

$$L_i^{class} = -\frac{1}{N}\sum_{i=1}^{N}(y_i^{class}\log(y_i^{predict}) + (1-y_i^{class})\log(1-y_i^{predict}))$$

$$y_i^{class} \in \{0,1\}$$

式中：y_i^{class} 为该形状框下的实际形状标签，$y_i^{predict}$ 为预测的概率。

所述步骤 2 中箭头指向对象估计具体为：

根据形状坐标检测输出的最终位置的形状框中形状类型识别为箭头/直线/双箭头的位置坐标框(统称箭头框)，独立地对每个箭头框的关键点进行特征提取和回归：

$(x_{begin}^{truth}, y_{begin}^{truth})$、$(x_{end}^{truth}, y_{end}^{truth})$ 为实际箭头框中的两个真实二维关键点：起始点与终止点，$(x_{begin}^{predict}, y_{begin}^{predict})$、$(x_{end}^{predict}, y_{end}^{predict})$ 为预测箭头框中的两个对应的预测二维关键点，箭头关键点线性回归方法通过计算实际箭头框中的真实二维关键点与预测箭头框中的对应的预测二维关键点之间的欧式距离，计算回归损失函

数

$$L_{ij}^{\text{keypoint}} = \sqrt{\| x_{ij}^{truth} - x_{ij}^{preidct} \|_2^2 + \| y_{ij}^{truth} - y_{ij}^{preidct} \|_2^2}$$

$$s.t. \begin{cases} i \in arrow \\ j \in [begin, end] \end{cases}$$

式中：$(x_{ij}^{truth}, y_{ij}^{truth})$ 为第 i 个实际箭头框中的真实二维关键点，$(x_{ij}^{predict}, y_{ij}^{predict})$ 为对应的第 i 个预测箭头框中的预测二维关键点。

所述根据箭头关键点检测输出的箭头框中的关键点的最终位置，与其周边的形状，根据形状位置、关键点位置以及像素坐标系下两者之间存在的几何关系，使用智能算法估计箭头关键点的归属关系，具体为：

将关键点与邻近形状对象之间的距离度量抽象为点到直线的欧式距离，对于给定多边形与箭头关键点的距离，使用点到直线的距离公式，选择最短距离为关键点到多边形的距离，表达式如下：

$$d_{\min} = \min_i | \frac{A_i x_0 + B_i y_0 + C_i}{\sqrt{A_i^2 + B_i^2}} |$$

$$s.t. \begin{cases} A_i = y_i^1 - y_i^2 \\ B = x_i^1 - x_i^2 \\ C = y_i^1 (x_i^1 - x_i^2) - x_i^1 (y_i^1 - y_i^2) \end{cases}$$

式中：(x_0, y_0) 为箭头关键点二维坐标，(x_i^1, y_i^1)、$(x_i^2,\ y_i^2)$ 为多边形第 i 边的两个端点坐标；

对于所有预测的形状对象与箭头关键点均计算出距离，选择最小距离的多边形为箭头关键点的归属对象，表达式如下：

$$S = \arg\min_i \ d_i$$

式中：d_i为箭头关键点到所有预测形状对象中的第i个形状的距离。

所述步骤 3 中绘图软件 PowerPoint / Visio 结合软件特定文档接口输入要求，提取并转化神经网络原始输出，编写接口函数自动生成预测形状，同时通过箭头指向对象估计，进而推断出对象之间的连接关系，起始点为因，终止点为果，编写函数实现形状对象间的关系连接具体为：

在与 PowerPoint 软件对接中，使用现有的 PowerPoint 开发工具：python 语言的 pptx 扩展包，通过直接操作 PowerPoint 软件并进行一系列添加形状、连接对象、设置文本操作；

在与 Visio 软件对接中，使用 python 语言的 win32com 扩展包，通过 Windows 操作系统层面启动 Visio 软件并进行一系列添加形状、连接对象、设置文本操作；

针对 pptx 与 win32com 可用接口要求，将神经网络的原始输出中的形状位置转化为中心点＋范围模式，矩形示例如下：

$$(x_0, y_0, y_1, y_2) \quad \rightarrow \quad (x_c, y_c, H, W)$$

$$s.t. \begin{cases} x_c = \dfrac{|x_0 + x_1|}{2} \\ y_c = \dfrac{|y_0 + y_1|}{2} \\ W = |x_0 - x_1| \\ H = |y_0 - y_1| \end{cases}$$

式中：(x_0, y_0, y_1, y_2)为神经网络原始坐标输出，(x_c, y_c)为矩形中心点，(H, W)为矩形长宽；

将神经网络的原始输出中的箭头关键点根据箭头对象估计所对应的所属形状对象转化为形状对象的连接属性选择，矩形示例如下：

$$(x_{begin}, y_{begin}) / (x_{end}, y_{end}) \quad \rightarrow \quad (center, down, right, up, left)$$

式中：$(x_{begin},y_{begin})/(x_{end},y_{end})$为箭头关键点的起始点/终止点，$(center,down,right,up,left)$为矩形对象的可选接口点；

同时，在 Visio 操作中，通过将上述接口调用封装为直接调用函数进行使用简化。

所述步骤 3 中整体轮廓智能排版具体为：

一次聚类：使用可编辑图形的长度与宽度为特征，使用 canopy 算法＋kmeans 聚类模式：先对所有可编辑图形进行粗聚类，设置两个阈值分别为T_1和T_2，再通过 canopy 算法，将长宽的特征距离小于阈值的视为相同大小，获得聚类数K与各个聚类中心m_i；再设置 kmeans 聚类算法的聚类数为K，初始聚类中心为m_i，使用 kmeans 聚类算法，得到聚类后的K个可编辑图形集合，取特征平均值作为类别中心，将同一类中的所有可编辑图形的大小设置为类别中心的大小，表达式如下：

$$\min \ J=\sum_{i=1}^{K}\sum_{x\in C_i}\| x-m_i \|^2$$

$$s.t.\ m_i=\frac{1}{N_i}\sum_{x\in C_i}x$$

式中：m_i为第i类数据的聚类中心，C_i为第i类数据集合；

二次聚类：使用可编辑图形的左上角横纵坐标为特征，使用 canopy 算法＋kmeans 聚类模式：先对所有可编辑图形进行粗聚类，设置两个阈值分别为T_1和T_2，再通过 canopy 算法，将坐标的特征距离小于阈值视为同一水平线/竖直线；获得聚类数K与各个聚类中心m_i；再设置 kmeans 聚类算法的聚类数为K，初始聚类中心为m_i，使用 kmeans 聚类算法，得到聚类后的K个可编辑图形集合，将同一类中的所有可编辑图形的对齐值设置为类别中心的对齐值，实现自动对齐，

表达式与上式相同；

同时提供Visio软件自带的智能排版算法，根据智能排版接口的输入要求，将箭头关键点因果转换的形状对象的连接属性选择进一步转化为自动连接属性选择，矩形示例如下：

$$(center,down,right,up,left) \rightarrow \begin{pmatrix} AutoConnectDirDown \\ AutoConnectDirUp \\ AutoConnectDirRight \\ AutoConnectDirLeft \end{pmatrix}$$

式中：$(center,down,right,up,left)$为矩形对象的可选接口点，$\begin{pmatrix} AutoConnectDirDown \\ AutoConnectDirUp \\ AutoConnectDirRight \\ AutoConnectDirLeft \end{pmatrix}$为自动连接属性可选接口点。

所述步骤4中OCR模块搭载具体为：

选用百度飞桨下的paddle paddle的自然语言处理包paddleOCR的python版本，对形状坐标检测输出的最终位置的形状框中形状类型识别为文本的位置坐标框进行截取，并使用 paddleOCR 进行识别，同时与预测其他形状的预测框进行重合度检测，表达式如下：

$$J_{IoU}(S_1,S_2)=\frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$$

式中：S_1为给定形状的预测框，S_2为文本的预测框；

对于J_{IoU}大于给定阈值的文本预测框，将其认定为相应形状预测框的内置文本；对于J_{IoU}小于给定阈值的文本预测框，将其认定为自由文本，同时与PowerPoint / Visio软件对接，将形状的内置文本嵌入到对应形状，在自由文本预测位置处生成文本框并嵌入相应文本。

本申请的另一发明目的，在于提供一种手绘场景下的图像识别与智能转化系统，包括：

存储器，用于存储可由处理器执行的指令；

处理器，用于执行所述指令以实现如上所述的方法。

本申请的另一发明目的，在于提供一种计算机可读介质，存储有计算机程序代码，计算机程序代码在由处理器执行时实现如上所述的方法。

本发明的有益效果：

本发明提出了手绘场景下的图像识别与智能转化方法、系统及计算机可读介质，综合考虑形状的位置与类别，形状通过指定箭头构成的相互关系，可实现箭头关键点因果转换，确定箭头指向对象，并开发了超轻量版的 Visio 软件 python 版本开发工具，为当前端到端的手绘流程图转化为特定软件的可编辑图形提供了解决方案。

本发明手绘场景下的图像识别与智能转化系统的各模块之间的功能相互独立，处理模块仅通过软件包形式嵌入计算机，无需额外底层软件或程序支持；系统可快速与计算机进行适配，无需特殊设备支持；执行软件功能性强，代码短小简洁，方便进行各类型安全审查。

附图说明

图 1 为本发明手绘场景下的图像识别与智能转化方法的工作流程图。

图 2 为本发明中手绘流程图示例图。

图 3 为本发明中形状坐标检测和形状类型识别效果实例图。

图 4 为本发明中箭头特征点定位效果示例图。

图 5 为本发明中箭头对象估计示意图。

图 6 为本发明中箭头关键点因果转换示意图。

图 7 为本发明中整体轮廓智能排版效果示例图。

图 8 为本发明中 OCR 模块识别文字效果示例图。

图 9 为本发明手绘场景下的图像识别与智能转化系统的系统架构图。

图 10 为本发明的一种优选实例效果示例图。

具体实施方式

下面结合附图对本发明作进一步详细说明。

如图 1 所示，本发明手绘场景下的图像识别与智能转化方法，包括以下步骤：

手绘草图采集：

通过相机实时拍摄，实时扫描当前的手绘草图，或直接软件绘制流程图，采集当前时刻的手绘流程图像信息,手绘流程图像信息包括相机拍摄的原始 RGB 图像、经过扫描软件二值化加工的预处理图像，或在电子设备上直接绘制生成的图片，如图 2 所示；

定位形状位置和识别形状类别，主要包括形状坐标检测、形状类型识别、箭头特征点定位、箭头指向对象估计、设置置信度阈值：

形状坐标检测：

将采集的原始手绘流程图像信息输入至深度级联神经网络模型中进行特征提取，然后进行形状候选框筛选， 输出每个形状的位置坐标框；

将采集的手绘流程图的图像通过形状/背景二分类器输出得到各个可能含有形状的位置坐标框，将得到的各个可能含有形状的位置坐标框输入至深度级联神经网络模型中的第一个子网络结构中进行特征提取，输出众多候选框，对输出的候选框使用边界框回归方法合并重叠的候选框，得到校正后的候选框； 校正后的候选框分为两部分，第一部分用来判断当前校正后的候选框内图像是否存在形状(本发明实施例第一部分输出向量大小为 1×1×2)，第二部分表示当前校正后的候选框内形状位置相对真实目标的形状框位置的偏移(本发明实施

例第二部分输出向量大小 1×1×4)；

其中，形状/背景二分类器，把采集的手绘流程图图片中各个可能含有形状的边界框分割出来，如图 3 所示；

其中，形状/背景二分类器的交叉熵损失函数L_i^{truth}，表达式如下：

$$L_i^{truth} = -(y_i^{truth}\log(p_i) + (1-y_i^{truth})(1-\log(p_i)))$$

$$y_i^{truth} \in \{0,1\}$$

式中：y_i^{truth}为形状/背景的实际标签，p_i为形状的概率；

其中，边界框回归方法使用非极大值抑制合并重叠的候选框，得到校正后的候选框，边界框回归方法通过计算候选框的背景坐标 $\hat{y}_i^{box}$ 与实际的背景坐标y_i^{box}之间的欧式距离，计算回归损失函数L_i^{box}，表达式如下：

$$L_i^{box} = \| \hat{y}_i^{box} - y_i^{box} \|_2^2$$

$$y_i^{box} \in R^4$$

式中：$\| \|_2^2$ 表示欧氏距离的 L2 损失函数，y_i^{box}表示第 i 个真实形状框的位置坐标，$\hat{y}_i^{box}$ 表示对应的第i个预测形状框的位置坐标，R^4表示四维实数空间。

其中，y是一个四元组，包括候选框左上角的横坐标、候选框左上角的纵坐标、候选框右下角的横坐标、候选框右下角的纵坐标，如表 1 给出了部分预测的形状位置坐标：

其中， 第一个子网络是一个多层卷积网络，其模型架构如表 1 所示：

表 1 第一个子网络的模型结构

Name	Kernel size	Stride	Padding
Conv1	1x1x256	1x1	1x1
Conv2	3x3x256	1x1	1x1
Conv3	1x1x512	1x1	1x1

Conv4	3x3x256	1x1	1x1
Conv5	1x1x1024	1x1	\
Conv6	3x3x256	1x1	1x1
Conv7	1x1x2048	1x1	\
Conv8	3x3x256	1x1	1x1

形状类型识别：

将通过第一个子网络得到校正后的候选框作为输入，通过深度级联神经网络模型中的第二个子网络结构，输出粗预测的候选框， 使用边界框回归方法合并重叠的粗预测的候选框，得到第二次校正后的候选框；

将通过第二个子网络结构得到的第二次校正后的候选框作为输入，通过深度级联神经网络模型中的第三个子网络结构，输出最终位置的形状位置坐标框。

将形状位置坐标框输入至深度神经网络模型中进行特征提取，然后进行形状类型识别，输出每个形状的位置坐标框内的形状类型，如图 3 所示；

通过第三个子网络结构得到的形状位置坐标框输入至深度神经网络模型中进行特征提取，通过形状多分类器输出得到该形状位置框内为各个形状的概率，将得到的各个形状所属的概率通过 softmax 层进行归一化

其中，softmax 层表达式所下：

$$p_i = \frac{e^{z_i}}{\sum_{j}^{N} e^{z_j}}$$

式中： z_i、z_j 为分类器原始输出， p_i 为归一化概率，N 为形状类别总数。

其中， z 是一个一元组，即网络针对候选框的原始输出；

其中，形状多分类器的交叉熵损失函数 L_i^{class}，表达式如下：

$$L_i^{class} = -\frac{1}{N}\sum_{i=1}^{N}(y_i^{class}\log(y_i^{predict}) + (1-y_i^{class})\log(1-y_i^{predict}))$$

$$y_i^{class} \in \{0,1\}$$

式中：y_i^{class} 为该形状框下的实际形状标签，$y_i^{predict}$ 为预测的概率，N 为形状类别总数；

其中，y 是一个 N 元组，包括候选框中的预测属于各个形状的概率，如表 2 给出了部分形状框的预测类型及概率：

其中，第二、三个子网络是多层卷积+全连接网络：

第二个子网络为 ResNet50 网络。

第三个子网络模型架构如表 2 所示：

表 2 第三个子网络的模型结构

Name	Kernel size	Stride	Padding
Conv1	3x3x256	1x1	1x1
ReLU	\	\	\
Conv2	1x1x3	1x1	\
Conv3	1x1x12	1x1	\
Cls_score	1024x13	\	\
Bbox_pred	1024x48	\	\

箭头特征点定位：

通过形状坐标检测输出的最终位置的形状框中形状类型识别为箭头/直线/双箭头的位置坐标框(统称箭头框)，独立地对每个箭头框的关键点进行特征提取和回归，具体为：

$(x_{begin}^{truth}, y_{begin}^{truth})$、$(x_{end}^{truth}, y_{end}^{truth})$ 为实际箭头框中的两个真实二维关键点：起始点与终止点，$(x_{begin}^{predict}, y_{begin}^{predict})$、$(x_{end}^{predict}, y_{end}^{predict})$ 为预测箭头框中的两个对应的预测二维关键点，箭头关键点线性回归方法通过计算实际箭头框中的真实二维关键点与预测箭头框中的对应的预测二维关键点之间的欧式距离，如图 4 所示。计算回归损失函数

$$L_{ij}^{\text{keypoint}} = \sqrt{\| x_{ij}^{truth} - x_{ij}^{preidct} \|_2^2 + \| y_{ij}^{truth} - y_{ij}^{preidct} \|_2^2}$$

$$s.t. \begin{cases} i \in arrow \\ j \in [begin, end] \end{cases}$$

式中：$(x_{ij}^{truth}, y_{ij}^{truth})$ 为第 i 个实际箭头框中的真实二维关键点，$(x_{ij}^{predict}, y_{ij}^{predict})$ 为对应的第 i 个预测箭头框中的预测二维关键点。

其中，箭头关键点检测网络是多层卷积网络，其模型架构如表 3 所示：

表 3 箭头关键点检测网络的模型结构

Name	Kernel size	Stride	Padding
Conv1	3x3x512	1x1	1x1
Conv2	3x3x512	1x1	1x1
Conv3	3x3x512	1x1	1x1
Conv4	3x3x512	1x1	1x1
Conv5	3x3x512	1x1	1x1
Conv6	3x3x512	1x1	1x1
Conv7	3x3x512	1x1	1x1
Conv8	3x3x512	1x1	1x1
ReLU	\	\	\
ConvTranspose	4x4x2	2x2	1x1

箭头指向对象估计：

根据箭头关键点检测输出的箭头框中的关键点的最终位置，与其周边的形状，根据形状位置、关键点位置以及像素坐标系下两者之间存在的几何关系，使用智能算法估计箭头关键点的归属关系，如图 5 所示；

将关键点与邻近形状对象之间的距离度量抽象为点到直线的欧式距离，对于给定多边形与箭头关键点的距离，使用点到直线的距离公式，选择最短距离为关键点到多边形的距离，如下式所示：

$$d_{\min} = \min_i |\frac{A_i x_0 + B_i y_0 + C_i}{\sqrt{A_i^2 + B_i^2}}|$$

$$s.t.\begin{cases} A_i = y_i^1 - y_i^2 \\ B = x_i^1 - x_i^2 \\ C = y_i^1(x_i^1 - x_i^2) - x_i^1(y_i^1 - y_i^2) \end{cases}$$

式中：(x_0, y_0)为箭头关键点二维坐标，(x_i^1, y_i^1)、$(x_i^2,\ y_i^2)$为多边形第i边的两个端点坐标；

对于所有预测的形状对象与箭头关键点均计算出距离，选择最小距离的多边形为箭头关键点的归属对象，如图 6 所示，计算式为：

$$S = \arg\min_i \ d_i$$

式中：d_i为箭头关键点到所有预测形状对象中的第i个形状的距离。

设置置信度阈值：

对输入的手绘流程图图片执行上述形状坐标检测、形状类型识别、箭头特征点定位、箭头指向对象估计，输出每个预测形状框下预测的各个形状的概率，选取形状框预测位置合理、框内形状预测概率高的计算结果作为最终的形状框位置坐标及形状框内类型输出结果。

PPT/Visio 软件智能展示，主要包括深度神经网络与软件接口实现、箭头关键点因果转换、整体轮廓智能排版：

深度神经网络与软件接口实现：

选定绘图软件 Visio，结合软件特定文档接口输入要求，提取并转化神经网络原始输出，编写接口函数自动生成预测形状；

由于没有合适的 Visio 开发工具，本发明使用 python 语言的 win32com 扩展包，通过 Windows 操作系统层面启动 Visio 软件并进行一系列添加形状、连

接对象、设置文本等操作。针对 win32com 可用接口要求，将神经网络的原始输出中的形状位置转化为中心点＋范围模式，矩形示例如下：

$$(x_0, y_0, y_1, y_2) \quad \rightarrow \quad (x_c, y_c, H, W)$$

$$s.t. \begin{cases} x_c = \dfrac{|x_0 + x_1|}{2} \\ y_c = \dfrac{|y_0 + y_1|}{2} \\ W = |x_0 - x_1| \\ H = |y_0 - y_1| \end{cases}$$

式中：(x_0, y_0, y_1, y_2) 为神经网络原始坐标输出，(x_c, y_c) 为矩形中心点，(H, W) 为矩形长宽。

箭头关键点因果转换：

通过箭头指向对象估计，进而推断出对象之间的连接关系，起始点为因，终止点为果，编写函数实现形状对象间的关系连接；

将神经网络的原始输出中的箭头关键点根据箭头对象估计所对应的所属形状对象转化为形状对象的连接属性选择，矩形示例如下：

$$(x_{begin}, y_{begin}) / (x_{end}, y_{end}) \quad \rightarrow \quad (center, down, right, up, left)$$

式中：$(x_{begin}, y_{begin}) / (x_{end}, y_{end})$ 为箭头关键点的起始点 / 终止点，$(center, down, right, up, left)$ 为矩形对象的可选接口点。

进而，可得在像素坐标系下，箭头对象估计所对应的所属形状对象与箭头关键点所属矩形的四条边连接关系转换运算模型：

$$index = \arg\min_i \left| \frac{(y_i^1 - y_i^2)x_0 + (x_i^1 - x_i^2)y_0 + y_i^1(x_i^1 - x_i^2) - x_i^1(y_i^1 - y_i^2)}{\sqrt{(y_i^1 - y_i^2)^2 + (x_i^1 - x_i^2)^2}} \right|$$

$$index \in (center, down, right, up, left)$$

$$(x_0, y_0) \in ((x_{begin}, y_{begin}), (x_{end}, y_{end}))$$

式中：(x_0, y_0)为箭头关键点二维坐标，(x_i^1, y_i^1)、$(x_i^2,\ y_i^2)$为矩形第i边的两个端点坐标。

整体轮廓智能排版：

进一步地，对于生成的原始可编辑图形，对比各个形状的绝对与相对位置，采用启发式聚类算法实现横向与纵向对齐；形状大小标准化、一致化等，完成自动智能排版，同时提供软件内置智能排版算法进行选择，具体为：

一次聚类：使用可编辑图形的长度与宽度为特征，使用 canopy 算法＋kmeans 聚类模式：先对所有可编辑图形进行粗聚类，设置两个阈值分别为T_1和T_2，再通过 canopy 算法，将长宽的特征距离小于阈值的视为相同大小，获得聚类数K与各个聚类中心m_i；再设置 kmeans 聚类算法的聚类数为K，初始聚类中心为m_i，使用 kmeans 聚类算法，得到聚类后的K个可编辑图形集合，取特征平均值作为类别中心，将同一类中的所有可编辑图形的大小设置为类别中心的大小，表达式如下：

$$\min\ \ J = \sum_{i=1}^{K} \sum_{x \in C_i} \| x - m_i \|^2$$

$$s.t.\ \ m_i = \frac{1}{N_i} \sum_{x \in C_i} x$$

式中：m_i为第i类数据的聚类中心，C_i为第i类数据集合。

二次聚类：使用可编辑图形的左上角横纵坐标为特征，使用 canopy 算法＋kmeans 聚类模式：先对所有可编辑图形进行粗聚类，设置两个阈值分别为T_1和T_2，再通过 canopy 算法，将坐标的特征距离小于阈值视为同一水平线/竖直线；获得聚类数K与各个聚类中心m_i；再设置 kmeans 聚类算法的聚类数为K，初始聚类中心为m_i，使用 kmeans 聚类算法，得到聚类后的K个可编辑图形集合，将

同一类中的所有可编辑图形的对齐值设置为类别中心的对齐值，实现自动对齐，如图 7 所示，表达式与上式相同。

在实际应用中，两次 canopy 算法的两个阈值会由算法内部的通用模型给出，如表 4 给出了通用的 canopy 算法阈值：

表 4 通用 canopy 算法阈值

canopy 算法	T_1	T_2
一次聚类	0	$\min\ (length_i^2 + width_i^2)/1.618$
二次聚类	0	$\min\ length/1.618$ $\min\ width/1.618$

式中：$(length_i, width_i)$ 为第 i 个形状的长宽，$(length, width)$ 为所有形状 的长宽。

同时提供 Visio 软件自带的智能排版算法，根据智能排版接口的输入要求，将箭头关键点因果转换的形状对象的连接属性选择进一步转化为自动连接属性选择，矩形示例如下：

$$(center, down, right, up, left) \rightarrow \begin{pmatrix} AutoConnectDirDown \\ AutoConnectDirUp \\ AutoConnectDirRight \\ AutoConnectDirLeft \end{pmatrix}$$

式中：$(center, down, right, up, left)$ 为矩形对象的可选接口点，$\begin{pmatrix} AutoConnectDirDown \\ AutoConnectDirUp \\ AutoConnectDirRight \\ AutoConnectDirLeft \end{pmatrix}$ 为自动连接属性可选接口点。

进而，箭头关键点因果转换的形状对象与自动连接属性存在一一对应关系：

$$\begin{aligned} down &\Leftrightarrow AutoConnectDirDown \\ up &\Leftrightarrow AutoConnectDirUp \\ right &\Leftrightarrow AutoConnectDirRight \\ left &\Leftrightarrow AutoConnectDirLeft \\ center &\Leftrightarrow AutoConnectDirAny \end{aligned}$$

式中：*AutoConnectDirAny* 为自动连接属性任意可选接口点。

进一步地，通过预留外部接口，实现与现成高质量 OCR 功能包的集成进行手绘流程图文字识别，并与软件接口对接，实现文字自动生成，如图 8 所示，具体为：

本发明选用百度飞桨下的 paddle paddle 的自然语言处理包 paddleOCR 的 python 版本，对形状坐标检测输出的最终位置的形状框中形状类型识别为文本的位置坐标框进行截取，并使用 paddleOCR 进行识别，相较于整图的总体 OCR，准确率显著提升，同时与预测其他形状的预测框进行重合度检测，表达式如下：

$$J_{IoU}(S_1,S_2)=\frac{|S_1\cap S_2|}{|S_1\cup S_2|}$$

式中：S_1 为给定形状的预测框，S_2 为文本的预测框。

进而，由于形状坐标检测输出均为矩形候选框，因此可得重合度检测具体计算公式为：

$$J_{IoU}=\frac{\|x_{text}^1-x_{text}^2\|y_{text}^1-y_{text}^2|\cap|x_{shape}^1-x_{shape}^2\|y_{shape}^1-y_{shape}^2\|}{\|x_{text}^1-x_{text}^2\|y_{text}^1-y_{text}^2|\cup|x_{shape}^1-x_{shape}^2\|y_{shape}^1-y_{shape}^2\|}$$

式中：(x_{text}^1,y_{text}^1)、$(x_{text}^2,\ y_{text}^2)$ 为文本框的左上坐标与右下坐标，$(x_{shape}^1,y_{shape}^1)$、$(x_{shape}^2,\ y_{shape}^2)$ 为形状候选框的左上坐标与右下坐标；

对于 J_{IoU} 大于给定阈值的文本预测框，将其认定为相应形状预测框的内置文本；对于 J_{IoU} 小于给定阈值的文本预测框，将其认定为自由文本。同时与 PowerPoint / Visio 软件对接，将形状的内置文本嵌入到对应形状，在自由文本预测位置处生成文本框并嵌入相应文本。

如图 1～图 10 所示，本发明实施例中，手绘场景下的图像识别与智能转化系统，包括采集模块、识别模块、转化模块和附加模块。采集模块由相机、扫描软件或电子设备等组成，与识别模块连接；识别模块与转化模块为安装在计算机

内的控制软件，其中，识别模块一端与采集模块连接，一端与转化模块连接；转化模块一端与识别模块连接，一端与绘图软件连接；附加模块为可选模块，一端与转化模块连接，一端与绘图软件连接；整个手绘场景下的图像识别与智能转化系统由计算机电源供电运行，无需二次标定。

图 9 展示了手绘场景下的图像识别与智能转化系统的实现架构，系统由四个模块组成，包括采集模块、识别模块、转化模块和附加模块；采集模块负责采集手绘流程图图片信息，手绘流程图图片信息包括相机拍摄的原始 RGB 图像、经过扫描软件二值化加工的预处理图像，或在电子设备上直接绘制生成的图片；识别模块对采集到的手绘流程图图片分别进行形状坐标检测、形状类型识别、箭头特征点定位、箭头指向对象估计等步骤，确定各个预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状；转化模块对识别模块的输出分别进行箭头关键点因果转换、整体轮廓智能排版等步骤，最终输出指定软件的可编辑图形；附加模块对识别模块的部分输出(识别为文本)进行文字识别，将文字识别结果嵌入转化模块的输出，最终输出完整的可编辑图形。

采集模块：如图 2 所示，由相机等普通摄影设备对手绘流程图拍摄、或通过扫描软件对手绘流程图进行扫描、或在电子设备上直接手绘流程图等，最后通过计算机与识别模块连接，实现实时的手绘流程图的采集与传输。

识别模块：作为计算机软件的一部分，包括形状坐标检测、形状类型识别、箭头特征点定位、箭头指向对象估计在内的分模块；通过获取采集模块输入的采集的手绘流程图图片信息，顺序流经各分模块，最终输出各个预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状等数据。

转化模块：作为计算机软件的一部分，包括箭头关键点因果转换、整体轮廓智能排版在内的分模块；通过获取识别模块输出的预测形状的位置坐标框、形状

类型、箭头关键点位置及箭头所属形状等数据，顺序流经各分模块，最终输出指定软件的可编辑图形。

附加模块：作为计算机软件的一部分，为该系统的可选模块；通过获取识别模块输出预测文本框的位置坐标，进行文字识别，将文字识别结果嵌入转化模块的输出，最终输出完整的可编辑图形。

上述的手绘场景下的图像识别与智能转化系统可以实施为计算机程序，保存在硬盘中，并可记载到处理器中执行，以实施本发明实施例的方法。

本发明实施例还提供了一种存储有计算机程序代码的计算机可读介质，所述计算机程序代码在由处理器执行时实现如上所述的基于深度神经网络的手绘场景下的图像识别与智能转化方法。

手绘场景下的图像识别与智能转化方法实施为计算机程序时，也可以存储在计算机可读存储介质中作为制品。例如，计算机可读存储介质可以包括但不限于磁存储设备（例如，硬盘、软盘、磁条）、光盘（例如，压缩盘（CD）、数字多功能盘(DVD)）、智能卡和闪存设备(例如，电可擦除可编程只读存储器(EPROM)、卡、棒、键驱动)。此外，本发明实施例描述的各种存储介质能代表用于存储信息的一个或多个设备和/或其它机器可读介质。术语“机器可读介质”可以包括但不限于能存储、包含和/或承载代码和/或指令和/或数据的无线信道和各种其它介质（和/或存储介质）。

应该理解，上述的实施例仅是示意。本发明描述的实施例可在硬件、软件、固件、中间件、微码或者其任意组合中实现。对于硬件实现，处理单元可以在一个或者多个特定用途集成电路（ASIC）、数字信号处理器（DSP）、数字信号处理设备（DSPD）、可编程逻辑器件（PLD）、现场可编程门阵列（FPGA）、处理器、控制器、微控制器、微处理器和/或设计为执行本发明所述功能的其它电子单元或

者其结合内实现。

需要说明的是，在本申请中，诸如第一、 第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来，而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。

本说明书中的各个实施例均采用相关的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。

以上所述仅为本发明的较佳实施例而已，并非用于限定本发明的保护范围。凡在本发明的精神和原则之内所作的任何修改、等同替换、改进等，均包含在本发明的保护范围内。

说明书附图

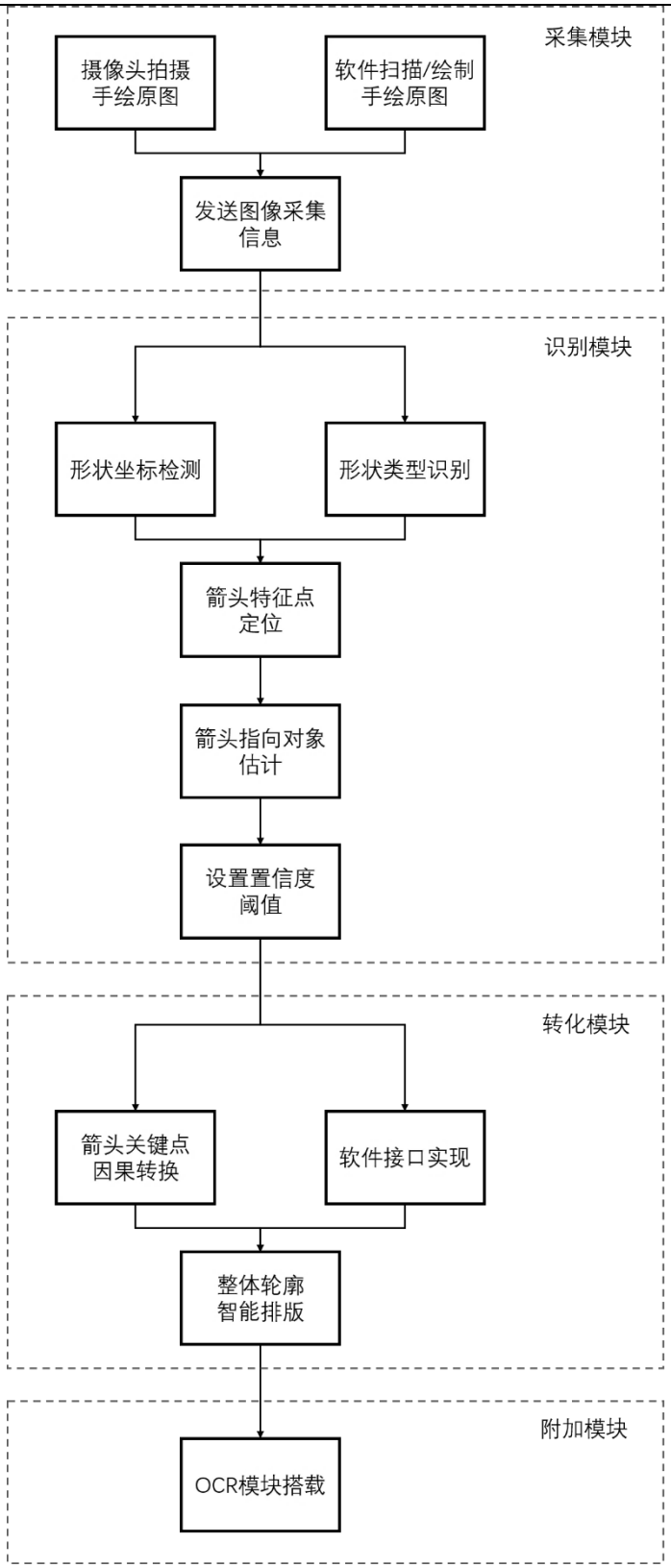
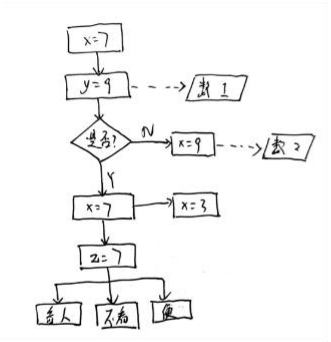
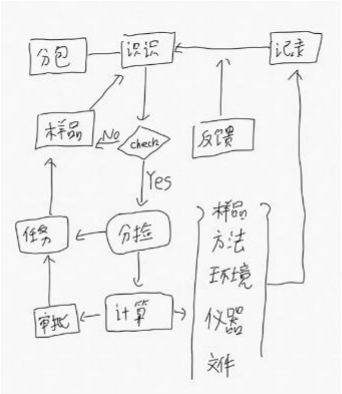

采集模块
摄像头拍摄
手绘原图
软件扫描/绘制
手绘原图
发送图像采集
信息
识别模块
形状坐标检测
形状类型识别
箭头特征点
定位
箭头指向对象
估计
设置置信度
阈值
转化模块
箭头关键点
因果转换
软件接口实现
整体轮廓
智能排版
附加模块
OCR模块搭载

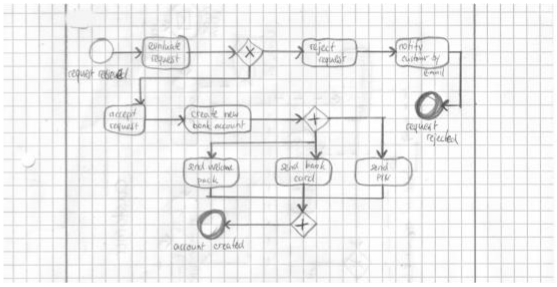
图 1



扫描图片



电子设备图片



拍摄图片

图 2

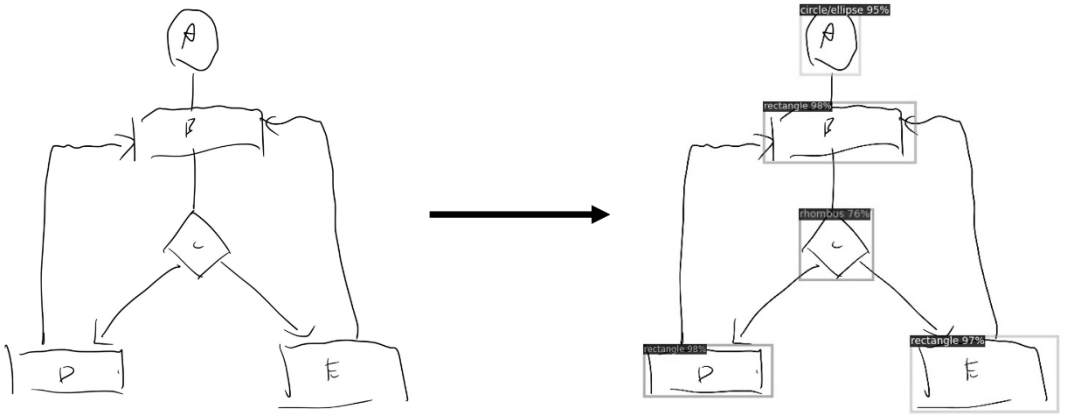


图 3

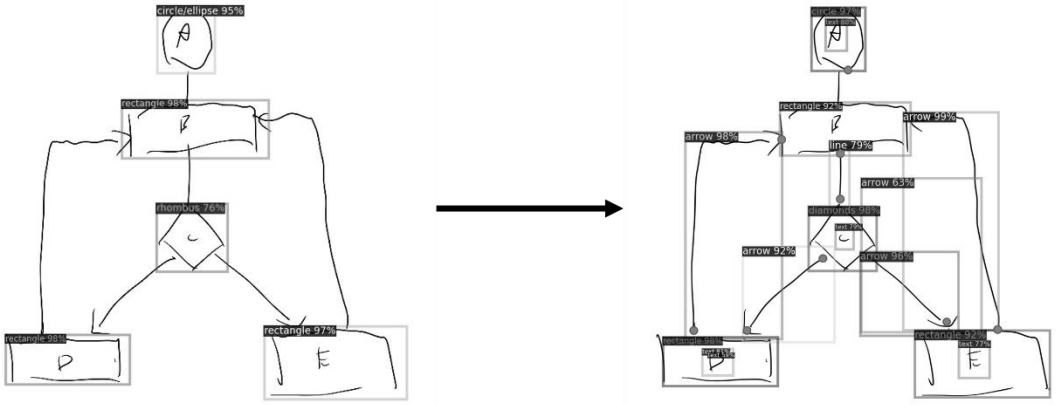
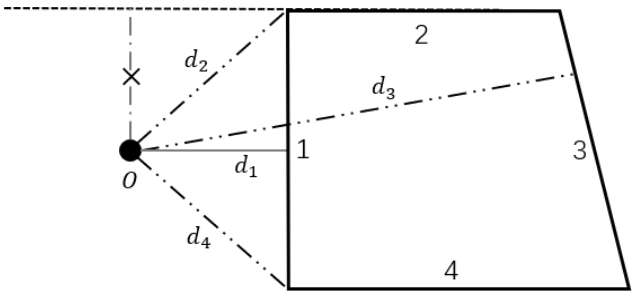
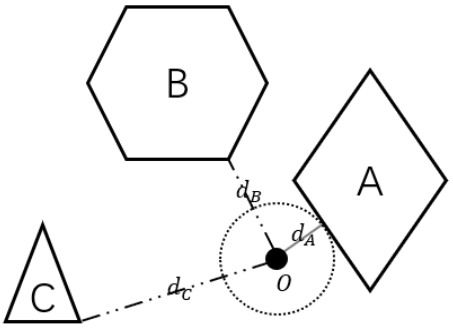


图 4



$d_O = d_1$



$O \in A$

图 5

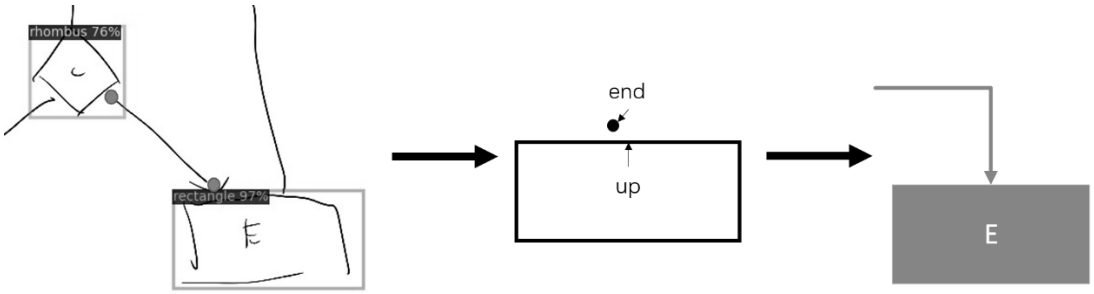


图 6

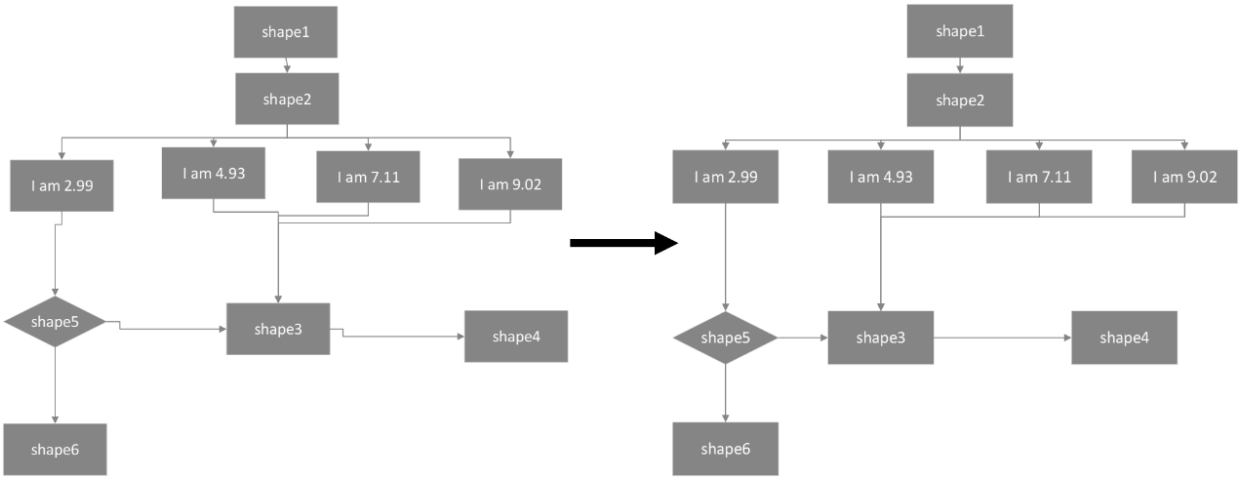
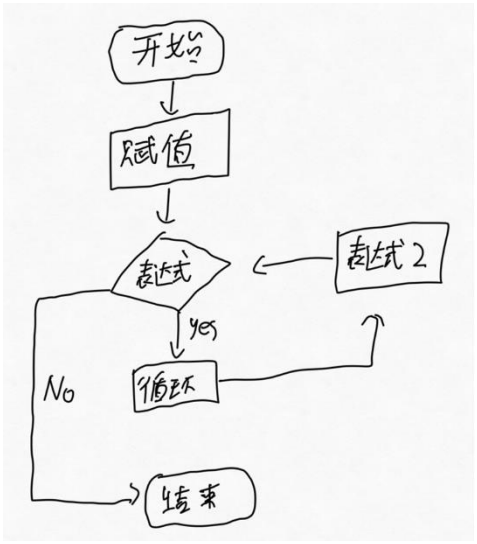


图 7



形状	文字
长椭圆1	开始
矩形1	赋值
菱形1	表达式
矩形2	表达式2
矩形3	循环
长椭圆2	结束
文本框1	yes
文本框2	No

图 8

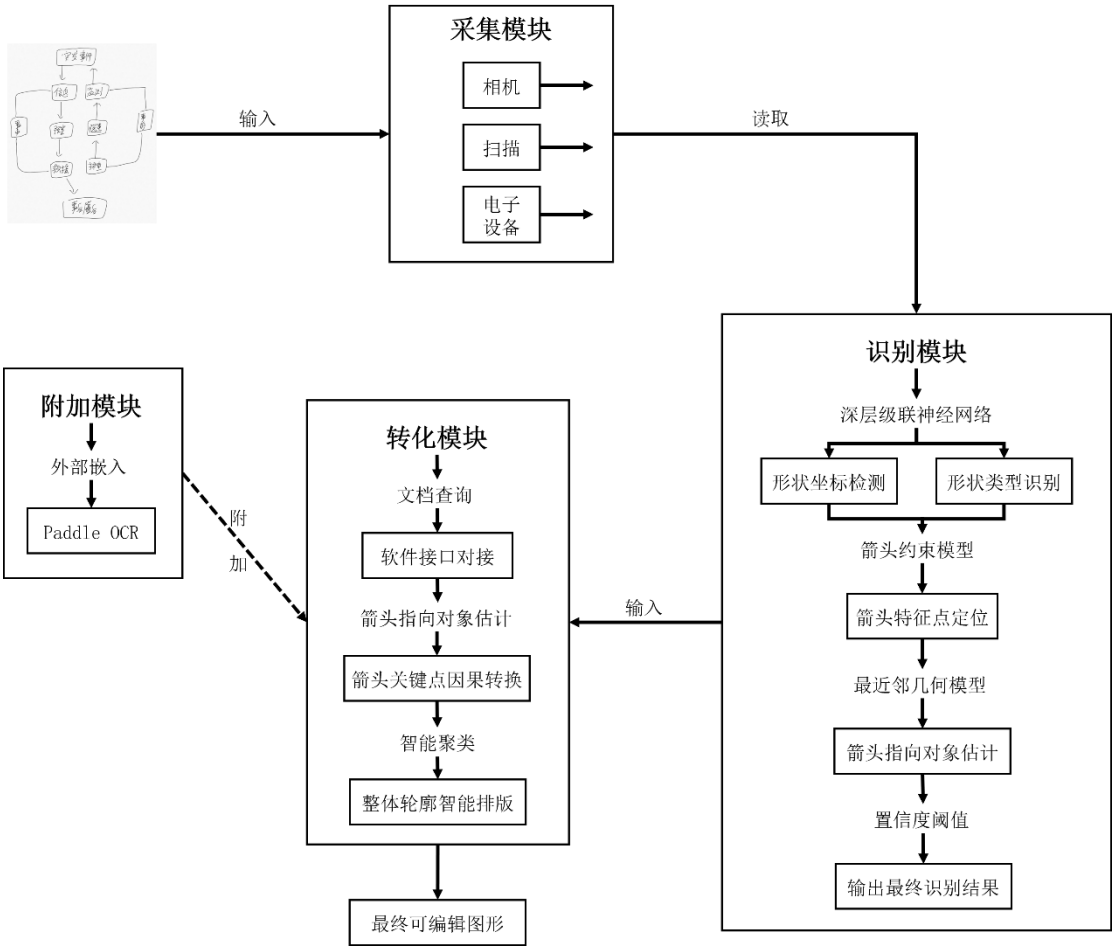


图 9

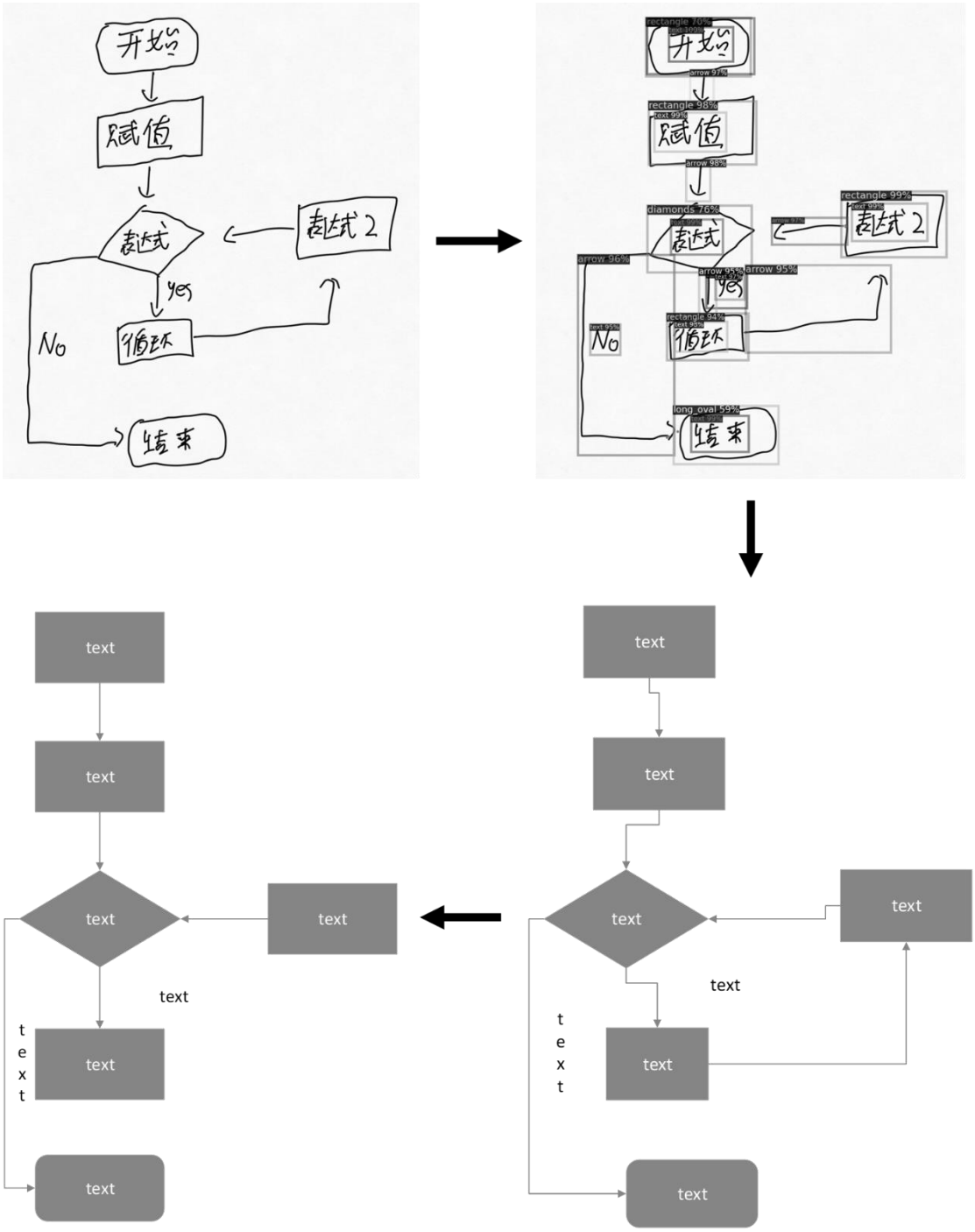


图 10

权利要求书

1.手绘场景下的图像识别与智能转化方法，其特征在于，包括以下步骤；

步骤 1：手绘流程草图采集：通过相机实时拍摄，实时扫描当前的手绘流程草图，或直接软件绘制流程图，采集当前时刻的手绘流程图像信息，输入至计算机，实现实时的手绘流程图采集与传输；

步骤 2：获取步骤 1 得到的手绘流程图像信息，通过定位形状位置和识别形状类别步骤，最终输出各个预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状数据；

步骤 3：获取步骤 2 的预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状数据，通过计算机绘图软件智能展示，输出指定软件的可编辑图形；

步骤 4：OCR 模块搭载，通过预留外部接口，实现与 OCR 功能包的集成进行手绘流程图文字识别，并与软件接口对接，将步骤 3 得到的指定软件的可编辑图形，实现文字自动生成，最终输出完整的可编辑图形。

2.根据权利要求 1 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 1 中手绘流程图像信息包括相机拍摄的原始 RGB 图像、经过扫描软件二值化加工的预处理图像，或在电子设备上直接绘制生成的图片。

3.根据权利要求 1 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 2 中定位形状位置和识别形状类别，包括形状坐标检测、形状类型识别、箭头特征点定位、箭头指向对象估计、设置置信度阈值；

形状坐标检测：将采集的原始手绘流程图像信息输入至深度级联神经网络模型中进行特征提取，然后进行形状候选框筛选，输出每个形状的位置坐标框；

形状类型识别：将所述每个形状的位置坐标框，并联神经网络模型中进行特

征提取，然后进行形状分类,结合形状位置坐标框输入至输出最终位置的形状框；

箭头特征点定位：将箭头转化为连接形状对象的因果关系，并使用始末特征点表示，通过约束模型对输出的最终位置的箭头形状框中的特征点进行检测，标注出最终位置的箭头形状框中的箭头二维特征关键点，实现箭头特征点定位；

箭头指向对象估计：对于输出的每一个箭头形状框中的特征点与其周边的形状，根据形状位置、关键点位置以及像素坐标系下两者之间存在的几何关系，使用智能算法估计箭头关键点的归属关系；

设置置信度阈值：对比各个预测形状在整体图像上的坐标位置及其置信度，结合箭头指向对象估计的确信度，选取形状位置合理、箭头指向对象明确，置信度高于设定阈值的识别结果作为最终的各个形状的坐标位置、类别及对象因果关系输出结果；

所述步骤 3 中计算机绘图软件智能展示，包括深度神经网络与软件接口实现、箭头关键点因果转换、整体轮廓智能排版；

深度神经网络与软件接口实现：结合计算机绘图软件特定文档接口输入要求，提取并转化神经网络原始输出，编写接口函数自动生成预测形状；

箭头关键点因果转换：通过箭头指向对象估计，进而推断出对象之间的连接关系，起始点为因，终止点为果，编写函数实现形状对象间的关系连接；

整体轮廓智能排版：对于生成的原始可编辑图形，对比各个形状的绝对与相对位置，采用启发式聚类算法实现横向与纵向对齐；形状大小标准化、一致化完成自动智能排版，同时提供软件内置智能排版算法进行选择。

4. 根据权利要求 3 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 2 中形状坐标检测具体为：

将采集的手绘流程图的图像通过形状/背景二分类器输出得到各个可能含

有形状的位置坐标框，将得到的各个可能含有形状的位置坐标框输入至深度级联神经网络模型中的第一个子网络结构中进行特征提取，输出众多候选框，对输出的候选框使用边界框回归方法合并重叠的候选框，得到校正后的候选框；校正后的候选框分为两部分，第一部分用来判断当前校正后的候选框内图像是否存在形状，第二部分表示当前校正后的候选框内形状位置相对真实目标的形状框位置的偏移；

将通过第一个子网络得到校正后的候选框作为输入，通过深度级联神经网络模型中的第二个子网络结构，输出粗预测的候选框，使用边界框回归方法合并重叠的粗预测的候选框，得到第二次校正后的候选框；

将通过第二个子网络结构得到的第二次校正后的候选框作为输入，通过深度级联神经网络模型中的第三个子网络结构，输出最终位置的形状位置坐标框；

5. 根据权利要求 3 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 2 中形状类型识别具体为：

通过第三个子网络结构得到的形状位置坐标框输入至深度神经网络模型中进行特征提取，通过形状多分类器输出得到该形状位置框内为各个形状的概率，将得到的各个形状所属的概率通过 softmax 层进行归一化，计算如下式所示：

$$p_i = \frac{e^{z_i}}{\sum_{j}^{N} e^{z_j}}$$

式中：z_i、z_j 为分类器原始输出，p_i 为归一化概率，N 为形状类别总数。

所述形状/背景二分类器的交叉熵损失函数 L_i^{truth}，表达式如下：

$$L_i^{truth} = -(y_i^{truth}\log(p_i) + (1-y_i^{truth})(1-\log(p_i)))$$

$$y_i^{truth} \in \{0,1\}$$

式中：y_i^{truth} 为形状/背景的实际标签，p_i 为形状的概率；

所述边界框回归方法使用非极大值抑制合并重叠的候选框，得到校正后的候选框，边界框回归方法通过计算候选框的背景坐标 $\hat{y}_i^{box}$ 与实际的背景坐标 y_i^{box} 之间的欧式距离，计算回归损失函数 L_i^{box}，表达式如下：

$$L_i^{box} = \| \hat{y}_i^{box} - y_i^{box} \|_2^2$$

$$y_i^{box} \in R^4$$

式中：$\| \|_2^2$ 表示欧氏距离的 L2 损失函数，y_i^{box} 表示第 i 个真实形状框的位置坐标，$\hat{y}_i^{box}$ 表示对应的第 i 个预测形状框的位置坐标，R^4 表示四维实数空间；

所述形状多分类器的交叉熵损失函数 L_i^{class}，表达式如下：

$$L_i^{class} = -\frac{1}{N}\sum_{i=1}^{N}(y_i^{class}\log(y_i^{predict}) + (1-y_i^{class})\log(1-y_i^{predict}))$$

$$y_i^{class} \in \{0,1\}$$

式中：y_i^{class} 为该形状框下的实际形状标签，$y_i^{predict}$ 为预测的概率。

6. 根据权利要求 3 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 2 中箭头指向对象估计具体为：

根据形状坐标检测输出的最终位置的形状框中形状类型识别为箭头/直线/双箭头的位置坐标框(统称箭头框)，独立地对每个箭头框的关键点进行特征提取和回归：

$(x_{begin}^{truth}, y_{begin}^{truth})$、$(x_{end}^{truth}, y_{end}^{truth})$ 为实际箭头框中的两个真实二维关键点：起始点与终止点，$(x_{begin}^{predict}, y_{begin}^{predict})$、$(x_{end}^{predict}, y_{end}^{predict})$ 为预测箭头框中的两个对应的预测二维关键点，箭头关键点线性回归方法通过计算实际箭头框中的真实二维关键点与预测箭头框中的对应的预测二维关键点之间的欧式距离，计算回归损失函

数

$$L_{ij}^{\text{keypoint}} = \sqrt{\| x_{ij}^{truth} - x_{ij}^{preidct} \|_2^2 + \| y_{ij}^{truth} - y_{ij}^{preidct} \|_2^2}$$

$$s.t. \begin{cases} i \in arrow \\ j \in [begin, end] \end{cases}$$

式中：$(x_{ij}^{truth}, y_{ij}^{truth})$ 为第 i 个实际箭头框中的真实二维关键点，$(x_{ij}^{predict}, y_{ij}^{predict})$ 为对应的第 i 个预测箭头框中的预测二维关键点；

所述根据箭头关键点检测输出的箭头框中的关键点的最终位置，与其周边的形状，根据形状位置、关键点位置以及像素坐标系下两者之间存在的几何关系，使用智能算法估计箭头关键点的归属关系，具体为：

将关键点与邻近形状对象之间的距离度量抽象为点到直线的欧式距离，对于给定多边形与箭头关键点的距离，使用点到直线的距离公式，选择最短距离为关键点到多边形的距离，表达式如下：

$$d_{\min} = \min_i \left| \frac{A_i x_0 + B_i y_0 + C_i}{\sqrt{A_i^2 + B_i^2}} \right|$$

$$s.t. \begin{cases} A_i = y_i^1 - y_i^2 \\ B = x_i^1 - x_i^2 \\ C = y_i^1 (x_i^1 - x_i^2) - x_i^1 (y_i^1 - y_i^2) \end{cases}$$

式中：(x_0, y_0) 为箭头关键点二维坐标，(x_i^1, y_i^1)、$(x_i^2,\ y_i^2)$ 为多边形第 i 边的两个端点坐标；

对于所有预测的形状对象与箭头关键点均计算出距离，选择最小距离的多边形为箭头关键点的归属对象，表达式如下：

$$S = \arg\min_i \ d_i$$

式中：d_i为箭头关键点到所有预测形状对象中的第i个形状的距离。

7.根据权利要求 3 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 3 中绘图软件 PowerPoint / Visio 结合软件特定文档接口输入要求，提取并转化神经网络原始输出，编写接口函数自动生成预测形状，同时通过箭头指向对象估计，进而推断出对象之间的连接关系，起始点为因，终止点为果，编写函数实现形状对象间的关系连接具体为：

在与 PowerPoint 软件对接中，使用现有的 PowerPoint 开发工具：python 语言的 pptx 扩展包，通过直接操作 PowerPoint 软件并进行一系列添加形状、连接对象、设置文本操作；

在与 Visio 软件对接中，使用 python 语言的 win32com 扩展包，通过 Windows 操作系统层面启动 Visio 软件并进行一系列添加形状、连接对象、设置文本操作；

针对 pptx 与 win32com 可用接口要求，将神经网络的原始输出中的形状位置转化为中心点＋范围模式，矩形示例如下：

$$(x_0,y_0,y_1,y_2) \quad \rightarrow \quad (x_c,y_c,H,W)$$

$$s.t.\begin{cases} x_c = \dfrac{|x_0+x_1|}{2} \\ y_c = \dfrac{|y_0+y_1|}{2} \\ W = |x_0-x_1| \\ H = |y_0-y_1| \end{cases}$$

式中：(x_0,y_0,y_1,y_2)为神经网络原始坐标输出，(x_c,y_c)为矩形中心点，(H,W)为矩形长宽；

将神经网络的原始输出中的箭头关键点根据箭头对象估计所对应的所属形状对象转化为形状对象的连接属性选择，矩形示例如下：

$$(x_{begin}, y_{begin})/(x_{end}, y_{end}) \quad \rightarrow \quad (center, down, right, up, left)$$

式中：$(x_{begin}, y_{begin})/(x_{end}, y_{end})$ 为箭头关键点的起始点/终止点，$(center, down, right, up, left)$ 为矩形对象的可选接口点；

同时，在 Visio 操作中，通过将上述接口调用封装为直接调用函数进行使用简化；

所述步骤 3 中整体轮廓智能排版具体为：

一次聚类：使用可编辑图形的长度与宽度为特征，使用 canopy 算法＋kmeans 聚类模式：先对所有可编辑图形进行粗聚类，设置两个阈值分别为T_1和T_2，再通过 canopy 算法，将长宽的特征距离小于阈值的视为相同大小，获得聚类数K与各个聚类中心m_i；再设置 kmeans 聚类算法的聚类数为K，初始聚类中心为m_i，使用 kmeans 聚类算法，得到聚类后的K个可编辑图形集合，取特征平均值作为类别中心，将同一类中的所有可编辑图形的大小设置为类别中心的大小，表达式如下：

$$\min \quad J = \sum_{i=1}^{K} \sum_{x \in C_i} \| x - m_i \|^2$$

$$s.t. \quad m_i = \frac{1}{N_i} \sum_{x \in C_i} x$$

式中：m_i 为第i类数据的聚类中心，C_i 为第i类数据集合；

二次聚类：使用可编辑图形的左上角横纵坐标为特征，使用 canopy 算法＋kmeans 聚类模式：先对所有可编辑图形进行粗聚类，设置两个阈值分别为T_1和T_2，再通过 canopy 算法，将坐标的特征距离小于阈值视为同一水平线/竖直线；获得聚类数K与各个聚类中心m_i；再设置 kmeans 聚类算法的聚类数为K，初始聚类中心为m_i，使用 kmeans 聚类算法，得到聚类后的K个可编辑图形集合，将

同一类中的所有可编辑图形的对齐值设置为类别中心的对齐值，实现自动对齐，表达式与上式相同；

同时提供Visio软件自带的智能排版算法,根据智能排版接口的输入要求，将箭头关键点因果转换的形状对象的连接属性选择进一步转化为自动连接属性选择，矩形示例如下：

$$(center,down,right,up,left) \rightarrow \begin{pmatrix} AutoConnectDirDown \\ AutoConnectDirUp \\ AutoConnectDirRight \\ AutoConnectDirLeft \end{pmatrix}$$

式中：$(center,down,right,up,left)$为矩形对象的可选接口点，$\begin{pmatrix} AutoConnectDirDown \\ AutoConnectDirUp \\ AutoConnectDirRight \\ AutoConnectDirLeft \end{pmatrix}$为自动连接属性可选接口点。

8.根据权利要求 3 所述的手绘场景下的图像识别与智能转化方法，其特征在于，所述步骤 4 中 OCR 模块搭载具体为：

选用百度飞桨下的 paddle paddle 的自然语言处理包 paddleOCR 的 python 版本，对形状坐标检测输出的最终位置的形状框中形状类型识别为文本的位置坐标框进行截取，并使用 paddleOCR 进行识别，同时与预测其他形状的预测框进行重合度检测，表达式如下：

$$J_{IoU}(S_1,S_2)=\frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$$

式中：S_1为给定形状的预测框，S_2为文本的预测框；

对于J_{IoU}大于给定阈值的文本预测框，将其认定为相应形状预测框的内置文本；对于J_{IoU}小于给定阈值的文本预测框，将其认定为自由文本，同时与PowerPoint / Visio 软件对接，将形状的内置文本嵌入到对应形状，在自由文本预测位置处生成文本框并嵌入相应文本。

9.手绘场景下的图像识别与智能转化系统，其特征在于，包括：

存储器，用于存储可由处理器执行的指令；

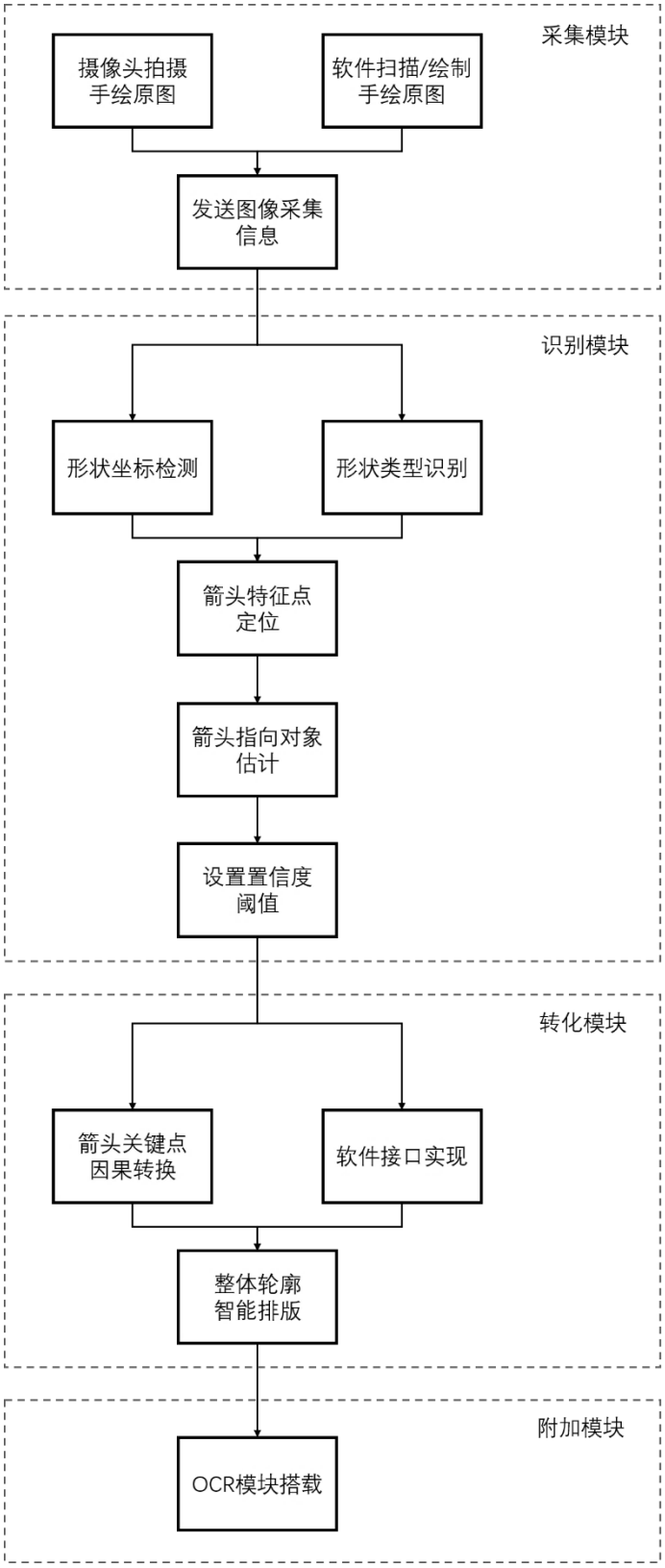
处理器，用于执行所述指令以实现如权利要求 1-8 任一项所述的方法。

10.一种计算机可读介质，其特征在于，存储有计算机程序代码，计算机程序代码在由处理器执行时实现如权利要求 1-8 任一项所述的方法。

说明书摘要

手绘场景下的图像识别与智能转化方法、系统及计算机可读介质，包括以下步骤；通过相机实时拍摄，实时扫描当前的手绘草图，或直接软件绘制流程图，采集当前时刻的手绘流程图像信息，输入至计算机，实现实时的手绘流程图的采集与传输；获取手绘流程图像信息，流入定位形状位置和识别形状类别中，最终输出各个预测形状的位置坐标框、形状类型、箭头关键点位置及箭头所属形状数据；通过 PPT/Visio 软件智能展示，输出指定软件的可编辑图形；OCR 模块搭载，实现文字自动生成，最终输出完整的可编辑图形。本发明解决了现有技术中存在的手绘流程图转化技术的研究的准确性、效率以及可用选择较低的问题。

摘要附图


采集模块
摄像头拍摄
手绘原图
软件扫描/绘制
手绘原图
发送图像采集
信息
识别模块
形状坐标检测
形状类型识别
箭头特征点
定位
箭头指向对象
估计
设置置信度
阈值
转化模块
箭头关键点
因果转换
软件接口实现
整体轮廓
智能排版
附加模块
OCR模块搭载