

THE CONCEPT OF ASSOCIATIVE GRAPHICAL INTERFACE IN THE WORKFLOW AUTOMATION SYSTEM

Abstract. The article is devoted to the topical problem of developing an associative graphical interface for workflow automation systems. Based on the analysis of modern technologies and research methods, the authors set a goal to develop a new interface concept that provides optimal efficiency and ease of use. The result is the creation of Draw & GO, a new tool for automating workflows. As part of the study, a plug in architecture was used, which makes it easy to integrate new functions and optimize the operation of the automation system. Key findings highlight the potential of the associative GUI in improving productivity and streamlining workflows.

Keywords: Draw & GO, automation, macros, performance, optimization, plugin architecture.

Statement of the problem. Information technology and computerized devices have become an integral part of the daily life of mankind and the organization of business processes. To save time and minimize monotonous work, an urgent task is to automate often repeated, identical, or similar actions. Automation software can help reduce human error, improve the efficiency of operations, increase speed, and free up time for more important tasks.

Currently, almost all users of computerized devices conduct a dialogue with the operating system using a graphical interface. The graphical interface has made file and device management easier by increasing the number of users and the number of applications for personal computers.

However, in our fast-paced present, the GUI is becoming more and more cumbersome and difficult to understand – it has many nested menu items and hidden settings. Therefore, a relevant area of research is the search for new ways of human-machine interaction.

Since modern devices are capable of perceiving a stylus, touch, and mouse movements, it becomes relevant to derive useful value from more than just clicks. One such method is the use of gesture or shape recognition, namely: the user draws a continuous shape, and the computer determines what has been done by matching

it with a predefined library of shapes. After the computer defines the form, any action can be specified for it.

Analysis of the latest research and publications. The workflow engine software market is highly competitive. The largest and most developed players in this market are Microsoft Flow [1] and WorkflowCore [2]. These two products are the de facto industry standards for workflow automation. Microsoft Flow is a product that allows users to automate workflows across multiple apps and services. While the service offers a wide range of features, it can be expensive and difficult to use for many users. As a result, there is a growing number of competitors offering similar services at a lower cost. WorkflowCore is an open-source workflow engine written in .NET Core, providing an easy way to implement complex workflows in any .NET application. While WorkflowCore is gaining popularity in the industry, other options offer different benefits: Make [3], Zapier [4], MacroMaker [5], JetBit MacroRecorder [6], etc. These programs allow users to set macros to make it easier to call programs.

The disadvantage of existing solutions is the lack of gesture-graphic control. Graphical control is currently used mainly only in applications and security shells. Well-known tools are Mazelock [7] and Windows Picture Password [8]. In addition to their advantages, these programs have a significant drawback – the limited gestures that are tied to certain shapes, that is, the user can guess the password after a while.

Objective. The research of this work is aimed at creating cross-platform software for automating workflows Draw & GO, which implements gesture-graphic interaction to achieve a minimum interface and maximum intuitiveness in control.

Presentation of the main material of the research. Draw & GO is an innovative cross-platform workflow system that is built on the microkernel architecture of plugins written on .NET MAUI and Blazor technologies, implements interaction based on graphic gesture recognition, allows users to create powerful and secure workflows customized according to the needs and preferences of each client based on specific conditions, actions and triggers in a secure environment of industry standards, Assists users in automating processes and numerous tasks. The software consists of a multi-step workflow that consists of user-defined plugins and connectors for connecting systems. The app combines workflow automation, visual design, and web-based APIs to provide a powerful and flexible solution for all users.

Draw & GO has several features that set it apart from other workflow mechanisms:

- Graphic Gesture Recognition: Draw & GO is one of the first workflow engines to use gesture recognition as an input method, users can draw any shapes on their screens to create workflows without typing or making mouse clicks;
- Visual Designer: Users can drag and drop plugins and connectors to define conditions, actions, triggers, loops, forks, and more;
- Real-time synchronization: users can see their changes instantly on all devices;
- Customization: Users can create their plugins or use existing ones from the library, including connectors that connect to a range of applications, including Microsoft, Google, and Dropbox.

The idea of using graphic gestures (Fig. 1) is based on a person's tendency to associative thinking [9]. The main role of associations in memorization is that we tie new knowledge to information we already know. The research of the German scientist G. Ebbinghaus shows that a person remembers objects and images better than sequences of symbols. In this way, personalized graphic code provides convenient and quick access to macros and Helps to optimize the workflow.



Figure 1 – Example of a user-drawn gesture to run a macro

To create and recognize graphic code, the DrawGo.GestureLibrary was created, which consists of the following 3 classes:

- PointPatternAnalyzer.cs – Performs an actual comparison of the gesture with a list of trained gestures.
- PointPatternMatchResult.cs – Represents the result of comparing a single gesture, including the macro ID and match probabilities.
- PointPatternMath.cs – contains helper functions for performing point distribution, angle calculation, and distance calculation (for point distribution, not comparison).

Before dividing, the gesture contains an array of unevenly spaced points and may contain more or less of the desired precision.

Let's take a look at the graphic gesture recognition algorithm, which is implemented in the PointPatternAnalyzer.GetPointPatternMatchResult (Macro compareTo, Point[] points method:

Let's break the graphic gesture into segments and find the desired length of each of them:

$$L_w = \frac{\sum_{i=1}^{n-1} \sqrt{(x_{i+1}-x_i)^2 + (y_{i+1}-y_i)^2}}{N}, \quad (1)$$

where n is the actual number of points, and N is the desired number of points. N has been chosen to be 100, but the more points we distribute for the gesture, the more accurate the comparison should be. However, the more points we check for each gesture, the slower the overall comparison will be. The distribution of a gesture implies the ability to significantly increase or decrease the number of points in a gesture to a certain fixed value. So, for example, a simple gesture like a straight line can be only 2 points, but a more complex gesture like M will take a minimum of 5 points. Most gestures will have many more points because they are made with a mouse or finger.

Let's set the initial values: the current length of the segment, the auxiliary point equals the first input point, which is also the first point we are looking for. $L_c = 0$

Let's set the boundaries of the indices $1 \leq i < n$ and $1 \leq j \leq N$

Let's set the current point as P_i

Let's find the increment of the current length of the segment as the distance between the auxiliary point and the current one:

$$\text{increment} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (2)$$

Let's find the test length of the segment:

$$l = L_c + \text{increment} \quad (3)$$

If $l < L_w$, then $L_c = l$, replace the auxiliary point with the current one, and go to step 4.

Let's calculate the proportionality factor to find a new point:

$$\text{koef} = \frac{L_w - L_c}{\text{increment}} \quad (4)$$

Let's find the coordinates of the desired point (Fig. 2)

$$X_j = (1 - \text{coef}_i) * x_{i-1} + \text{coef}_i * x_i \quad (5)$$

$$Y_j = (1 - \text{coef}_i) * y_{i-1} + \text{coef}_i * y_i \quad (6)$$

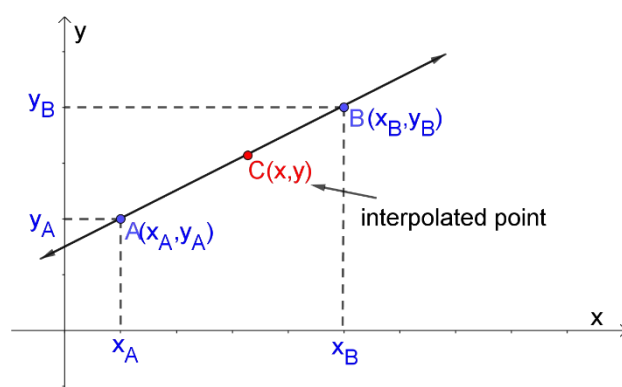


Figure 2 – Search for an interpolated point

If $j < N$, set the auxiliary point to the desired one, $L_c = 0$, reduce the index "i" by 1 and return to step 4.

The distribution is complete.

Once distributed, the movement shapes are preserved, and all points are evenly distributed and contain the desired number of points to compare.

When the distribution is complete, we convert the array of points into an array of angles. We do this because we are comparing the angular difference between the two gestures. We get an array of angles for each gesture.

$$\text{angle}_j = \tan^{-1}(Y_{j+1} - Y_j, X_{j+1} - X_j) \quad (7)$$

Now that we have angles for each gesture, we can iterate over each corner in the gesture and calculate the angular difference between the two corners. Let's place them in the created array.

$$\text{angleDifferences}_j = |\text{angle}_{j+1} - \text{angle}_j| \quad (8)$$

If $\text{angleDifferences}_j > \pi$

$$\text{angleDifferences}_j = \pi - (\text{angleDifferences}_j - \pi) \quad (9)$$

Use an array to calculate the mean angular difference between the two gestures, and convert it to a match accuracy (from 0% to 100%).

$$\text{probability} = 100 - \frac{\sum_{i=1}^n \text{angleDifferences}_i}{n} * \frac{100}{\pi} \quad (10)$$

That's all it takes to accurately recognize the difference between the two gestures. Once we have our probability, let's put everything in a PointPatternMatchResult, which contains the gesture and probability ID.

The above is a comparison of an array of points with a single pattern for a list of saved gestures. The next step is a calculation of the probability for each gesture in the sample set, which will help to evaluate them for the best match based on the highest probability.

Draw & GO is designed to be powerful and intuitive, providing a drag-and-drop-based graphical user interface for developing complex workflow systems (Figure 3).

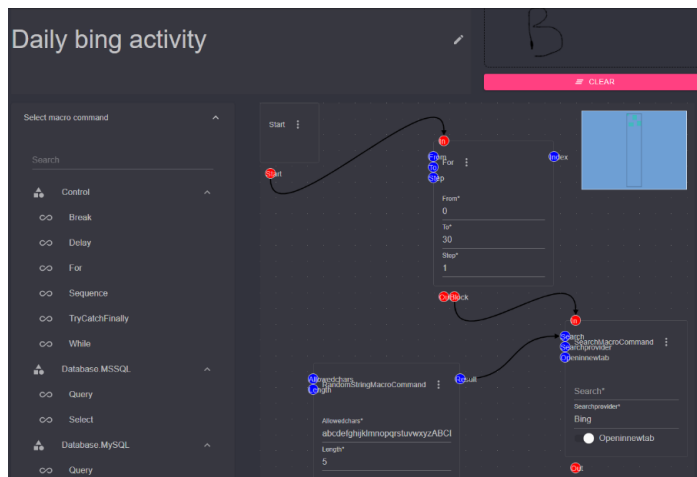


Figure 3 – Macro Configurator

Developers can create and customize flowcharts using drag-and-drop components, and the platform allows them to create customized applications that meet their specific business needs.

The software product "Draw & GO" consists of two modules, one of which is used to create and organize macros, and associate them with graphic code, and the second is used to open any exported macro and conduct online consultations. The created program allows users to increase the speed of working with a computer, reproducing several identical actions to facilitate the user's work and to protect private data. The "Give-a-Hand" subroutine allows users to organize online consultations, control another computer, and share files remotely.

Conclusions. Draw & GO is an innovative cross-platform workflow engine with microkernel plugin architecture and graphic gesture recognition. The mi-

crokernel architecture provides scalability, and plugin support allows users to customize the platform to suit their needs. With an easy-to-use visual workflow designer, users can quickly create and override complex workflows for many purposes. Regardless of what needs to be automated, whether user processes are simple or company workflows are complex, Draw & GO can help everyone achieve their goals. Due to the flexibility of its plugin architecture, it can be integrated with any existing third-party services such as Slack, Google Calendar, Dropbox, and many more. Draw & GO is designed to run on all major operating systems, including Windows, MacOS, iOS, Android, and Tizen. It works effectively on desktop and mobile devices, as well as web browsers, providing an intuitive user experience.

Users can get acquainted with the functionality of the Draw & GO software product and find out about available licenses on the official page <https://drawgo.azurewebsites.net> [10].

REFERENCES

1. Power Automate [Electronic resource]. – Access Mode: <https://powerautomate.microsoft.com/en-us/>.
2. Workflow Core [Electronic resource]. – Access Mode: <https://workflow-core.readthedocs.io/en/latest/>.
3. Make [Electronic resource]. – Access Mode: <https://www.make.com/en>.
4. Zapier [Electronic resource]. – Access Mode: <https://zapier.com/>.
5. MacroMaker [Electronic resource]. – Access Mode: <https://macromaker.en.softonic.com/>.
6. JitBit Macro Recorder [Electronic resource]. – Access Mode: <https://www.jitbit.com/macro-recorder/>.
7. Mazelock [Electronic resource]. – Access Mode: <https://eusing.com/mazelock/pclock.htm>.
8. Gao, H., Jia, W., Liu, N., Li, K. (2013). The Hot-Spots Problem in Windows 8 Graphical Password Scheme. In: Wang, G., Ray, I., Feng, D., Rajarajan, M. (eds) Cyberspace Safety and Security. CSS 2013. Lecture Notes in Computer Science, vol 8300. Springer, Cham. https://doi.org/10.1007/978-3-319-03584-0_26.
9. Antonyuk V.A., Segeda N.E. (2015). Recognition of graphical passwords as the means to accelerate the work with software projects. // Modern scientific and technical research in the context of language space. - T.2 – C. 5-6.
10. Draw & GO [Electronic resource]. – Access Mode: <https://drawgo.azurewebsites.net>

Received 21.11.2023.
Accepted 24.11.2023.

**Концепція асоціативного графічного інтерфейсу
у системі автоматизації робочих процесів**

Стаття акцентує увагу на важливій проблематиці створення інтуїтивно зрозумілих та ефективних інструментів для оптимізації та автоматизації робочих процедур. У контексті швидкого розвитку сучасних технологій автори статті визначили як свою мету розробку інноваційного інтерфейсу, який би забезпечував високу продуктивність з мінімальними зусиллями з боку користувачів.

Автори статті детально пропрацювали концепцію асоціативного графічного інтерфейсу, розглянувши глибину асоціативного мислення як фундаментального аспекту людської когніції. Це дозволило авторам сформулювати концепцію асоціативного графічного інтерфейсу, яка лягла в основу розробки інструменту "Draw & GO".

"Draw & GO" являє собою новаторське рішення, яке дозволяє користувачам швидко та зручно створювати автоматизації для повторюваних робочих процесів. Використовуючи метод *drag-and-drop*, користувачі можуть перетягувати макроси та створювати сценарії автоматизації без необхідності написання складного коду або розуміння принципів програмування.

Однією з ключових характеристик "Draw & GO" є його плагін-орієнтована архітектура. Архітектура плагінів надає велику гнучкість системі, дозволяючи легко додавати нові функціональні можливості та вдосконалюючи вже існуючі без перешкод для загальної ефективності системи. Такий підхід також забезпечує зниження часу, необхідного для впровадження нововведень та оптимізації робочих процесів завдяки швидкому розгортанню нових функцій. Draw & GO" ефективно працює на настільних та мобільних пристроях, а також у веб-браузерах, надаючи інтуїтивно зрозумілий користувацький досвід.

Користувачі можуть ознайомитися з функціональністю програмного продукту "Draw & GO" на офіційній сторінці <https://drawgo.azurewebsites.net>. Висновки дослідження подають "Draw & GO" як зразковий приклад синергії асоціативного графічного інтерфейсу та плагін-орієнтованої архітектури, що в результаті пропонує масштабоване, гнучке та користувач-орієнтоване рішення для автоматизації робочих процесів у різноманітних секторах бізнесу..

Ключові слова: Draw & GO, автоматизація, макроси, продуктивність, оптимізація, плагін-архітектура

Антонюк Владислав Андрійович - аспірант Дніпровського національного університету імені Олеся Гончара.

Сидорова Марина Геннадіївна - доцент Дніпровського національного університету імені Олеся Гончара.

Antonyuk Vladislav - PhD student Oles Honchar Dnipro National University.

Sydorova Maryna - Associate Professor Oles Honchar Dnipro National University.