

# Нейромережеве моделювання процесів розпізнавання графічних схем

Є.М. Чічірін, К.П. Сосненко

Інститут кібернетики імені В.М. Глушкова НАН України, 03187, м. Київ, проспект Академіка Глушкова, 40, chenevgen@ukr.net, sosnenkokate@ukr.net

*E.N. Chichirin, E.P. Sosnenko*

## NEURAL NETWORK MODELING OF GRAPHIC RECOGNITION PROCESSES

**Abstract.** Recognition of the structures of graphic images is impossible without identifying the relationships between their constituent objects. Algorithm flowcharts, along with simple short connections, can contain numerous tracing options for entangled relationships between shapes. In the case of direct neural network simulation of block diagrams together with all their connections, the number of outputs and network size would be unrealistically large for modern technologies. Thus, the final analysis of the relationships and the final structure of the flowcharts is proposed to be carried out on the basis of the rules of grammatical analysis of the relationships of the set of primary objects - graphic primitives of the BS image. For the sake of preserving unity in modelling and recognizing the lines of the contours of figures and the lines of connections between them, the nodes of the joints of the segments of the lines of the contours and their connections were selected as primary primitives. These are elementary nodes of the connection of line segments and lines with arrows, which depict turns, reproduction, convergence, and simply the continuation of straight and curved lines. A neural network model has been developed for recognizing the graphic listed primitives - nodes as part of the tools for parsing graphic images of flowcharts. A basic set of training and test images of BS nodes has been formed. The stages of the software synthesis of training data, as well as the augmentation of synthetic training data for their reproduction, are considered. It is noted that it is necessary not only to classify the node, but also to localize its position, since the reliability of the successful detection of each next node is determined by the accuracy of the combination of the center of the scan window and the center of the real node. Neural network models are proposed to increase the accuracy of detection of flowchart nodes based on fully accessible and convolutional neural networks without layers of subsampling. The expediency of combining syntactic and neural network recognition methods for a wide class of graphic images is substantiated.

**Key words:** artificial neural networks, machine learning, image recognition, parsing.

**Анотація.** Розроблена нейромережева модель розпізнавання графічних примітивів – вузлів у складі засобів синтаксичного аналізу зображень блок-схем. Сформована базова множина навчальних зображень вузлів БС.

*Розглянуто етапи програмного синтезу навчальних даних, а також аугментації синтетичних навчальних даних. Запропоновано нейромережеві моделі для підвищення точності детектування вузлів БС на основі повноз'язаних мереж та загорткових нейронних мереж без шарів субдискретизації.*

**Ключові слова:** штучні нейронні мережі, машинне навчання, розпізнавання зображень, синтаксичний аналіз.

**Анотація.** Разработана нейросетевая модель распознавания графических примитивов – узлов в составе средств синтаксического анализа изображений блок-схем. Сформировано базовое множество обучающих изображений узлов БС. Рассмотрены этапы программного синтеза обучающих данных, а также аугментации синтетических учебных данных. Предложены нейросетевые модели для повышения точности детектирования узлов БС на основе полносвязных сетей и сверточных нейронных сетей без слоев субдискретизации.

**Ключевые слова:** искусственные нейронные сети, машинное обучение, распознавание изображений, синтаксический анализ.

**Вступ.** Сучасні машини успішно досягли 99 % точності в визначенні ознак і об'єктів на зображеннях. Штучні нейронні мережі (ШНМ) імітують людський мозок для вирішення складних проблем і пошуку закономірностей в даних. За останні кілька років вони замінили багато алгоритмів машинного навчання і комп'ютерного зору. Водночас алгоритми візуального розпізнавання тільки нещодавно стали навчатися виявляти закономірності взаємодії та відносин різних об'єктів на зображенні (аналіз сцен, анотації зображень тощо).

Нейромережеве моделювання графічних зображень, наприклад, блок-схем (БС), належить до таких підрозділів машинного навчання і розпізнавання, яке теж не можливе без виявлення зв'язків між складовими фігурами БС. В роботі [1] зазначається, що аналіз структур БС можливий на рівні відрізків ліній і їх з'єднань (вузлів), або первинних об'єктів-фігур, та знову ж таки, ліній і їх з'єднань, що пов'язують ці фігури. У разі прямого нейромережевого моделювання

присутніх в обох випадках численних варіантів зв'язків фігур БС, кількість виходів і розмір ШНМ, що моделює БС, були б нереально великими для сучасних технологій.

Наприклад, при загальній кількості фігур БС  $m = 9$ , їх виходів  $n = 11$  (дві фігури предиката), кількість класів структур, і відповідно виходів ШНМ  $k = (m-1)^{n-1} = 2^{30} > 10^9$ , що в мільйон разів перевищує 1000 класів відомої мережі ImageNet. Кількість навчальних прикладів з урахуванням їх варіацій для кожного класу – порядку  $10^{11}$ . Таким чином, остаточний синтаксичний аналіз підсумкової структури БС поки що слід виконувати на основі правил граматичного розбору первинних двовимірних об'єктів (графічних примітивів) зображення БС, а не за допомогою ШНМ.

Але, якщо відволіктися від необхідності 100 % покриття варіантів зв'язків БС і враховувати лише реальну семантичну складову кожної БС хоча б на рівні сучасного LSA, ШНМ вже нині могли б наробити шуму.

Що стосується первинних примітивів зображень БС, різноманітність яких доступна для моделювання і розпізнавання за допомогою як ШНМ, так і синтаксичних методів, то в їхній якості можуть виступати:

- навіть пікселі (точки);
- довільно орієнтовані відрізки ліній контурів і зв'язків (як в [1]);
- вузли з'єднань відрізків ліній контурів і зв'язків (як пропонується);
- контури фігур БС (але тоді залишаються ще зв'язки між фігурами).

Для збереження єдності при моделюванні і розпізнаванні ліній контурів фігур і ліній з'єднань фігур як первинні примітиви обрано вузли з'єднань відрізків ліній контурів, зв'язків і нейромережеву згорткову модель для їх розпізнавання [2]. Згорткова нейронна мережа (ЗНМ) працює добре не тільки з зображеннями, але і з звуком, і текстами за рахунок істотно зменшеної кількості навчальних параметрів (вагових коефіцієнтів), які враховують лише суттєво корельовані (зазвичай, близькі в просторі, часі, застосуванні) елементи вхідних векторів. Сучасні фреймворки, такі як Tensorflow, PyTorch, Keras спрощують глибоке навчання ЗНМ.

**Синтез і аугментація даних.** Основні складності в разі нейромережевого розпізнавання переносяться на підготовку навчальних і

тестових (далі навчальних) множин прикладів (даних) і частково на досить тривалий процес машинного навчання. Розглянемо основні вимоги до вхідних і вихідних навчальних даних ЗНМ, які накладають на них остаточний синтаксичний підсумковий розбір структури БС.

Запропонований в [1] спосіб розпізнавання графічних зображень БС, включаючи зв'язки між фігурами, оснований на послідовному скануванні зображень вузлів вздовж ліній зв'язків і контурів фігур, зрівнянні променів  $L_i$  вузлів, у тому числі двопробових вузлів прямих ліній, з множиною  $V$  заздалегідь синтезованих еталонних радіус-векторів  $R_i$  з різною орієнтацією  $\varphi_i$  у площині графічних зображень, кратній постійному кроку  $\Delta\varphi = 2\pi/k$  їх кута повороту щодо деякого початкового кута  $\varphi_0$ :

$$V = \{R_i\}, \quad R_i = \{(x_i^v, y_i^v) \mid v = \overline{1, r}\},$$

$$\varphi_i = 2\pi * i / k = 360 * i / k, \quad i = \overline{0, k-1},$$

де  $(x_i^v, y_i^v) \in R_i$ , – локальні координати  $r$  точок радіус-вектора  $R_i$ .

Наявність променя  $L_i$  зображення в напрямку  $i$ -го радіус-вектора оцінюється шляхом зрівняння з деяким порогом суми бінарних значень пікселів зображення по абсолютним координатам цього радіус-вектора  $X = C_x + x_i^v$ ,  $Y = C_y + y_i^v$ , де  $C_x, C_y$  – координати центру сканування поточного вузла.

Класифікація вузлів за типами реалізується за допомогою сигнатури, в якості якої використовується зростаюча послідовність  $z_q^p = i_0 i_1 \dots i_p$ ,  $p < k$  індексів променів  $L_i$   $q$ -го вузла, відносно початкового (при скануванні) променя входу  $L_0$  в вузол  $q$ , в якій замість постійно нульового значення індексу  $i_0 = 0$  заноситься його абсолютне значення  $i_0 = \varphi_0 / \Delta\varphi$  щодо осі  $Ox$ .

Відповідно до прийнятих в [1] обмежень, зображення БС є частково орієнтований зв'язаний навантаженим текстом граф з множиною вершин  $\Phi$  і ребер  $E$ . Ребра графа утворюють замкнуті, можливо дугоподібні, контури фігур БС і незамкнуті (деревовидні) об'єднання ліній

зв'язку між фігурами, вершини – вузли з'єднань ліній контурів фігур і ліній зв'язку.

Сигнатури вузлів і замкнутість виключно контурів фігур є база для побудови рекурсивного алгоритму просування сканера вздовж ліній графа БС. При цьому відносність індексів сигнатур, крім початкового, робить алгоритм просування інваріантним до орієнтації променів входу  $L_0$  у вузли, а також дає можливість формування абсолютних координат наступних центрів сканування і атрибутів фігур БС як за поточними, так і за відкладеними в стек сигнатурами вузлів.

З зазначеного можна зробити такий висновок. Для сумісності існуючого алгоритму обходу контурів фігур і гілок зв'язків з запропонованим

нейромережовим розпізнаванням вузлів входом і виходом ШНМ на етапі навчання повинні бути графічні зображення і відповідні сигнатури для кожного класу можливих вузлів БС для обраного значення кутового кроку  $k$ .

Підготовка набору навчальних зображень вузлів (НЗВ) трудомісткий процес, особливо, якщо мова йде про реальні дані. З урахуванням того, що реальні БС в кінцевому підсумку готуються із застосуванням програмних засобів, в таблиці представлено такий програмно генерований базовий набір НЗВ вузлів БС із зазначенням їх класу та сигнатур для  $k = 24$  і  $\Delta\varphi = \pi/12 = 15^\circ$ . В таблиці 4 типи вузлів утворюють 131 клас, кожний з яких має унікальну сигнатуру.

ТАБЛИЦЯ. Базовий набір навчальних зображень вузлів БС

Тип вузла	Клас	Базові зображення можливих вузлів БС	Сигнатури
1-променеві	1...4		18 18^ 0 0^
2-променеві	5...64		0,12 0,11 0,6 14,22 2,10 1,13 1,12 6,12 ..... 2,14 ..... 12,18 17,19 5,7 ..... 12,23 0,18 9,21 0,13 4,18 8,18 10,22 ..... 11,23 10,23 1,23 1,13
3-променеві	65...124		6, 12^, 18^ ..... 6, 12, 18 0^, 12, 18^ ..... 0, 12, 18 0^, 6^, 18^ ..... 0, 6, 18 0, 6^, 12^ ..... 0, 6, 12  0, 7, 17...1, 8, 18 0^, 7, 17.. 0, 8, 16...1, 9, 17 0^, 8, 16.. 0, 9, 15...1, 10, 16 0^, 9, 15.. 0, 10, 14...1, 11, 15 0^, 10, 14.. 0, 11, 13... 1, 12, 14 0^, 11, 13..
4-променеві	125...131		0^, 6, 12^, 18 0, 6^, 12, 18^ 0^, 6^, 12^, 18 0, 6^, 12^, 18^ 0^, 6, 12^, 18^ 0^, 6^, 12, 18^ 0, 6, 12, 18

Формат представлення НЗВ  $m \times m = 21 \times 21$  пікселів. Значення формату  $m$  і кутового кроку  $k$  обрано з міркувань фільтрування впливу периферійних областей вузлів БС на помилки їх класифікації і збереження важливої інформації з навколо центральних областей цих вузлів. Значення індексів сигнатур  $1 \leq i \leq 127$  представлено байтом цілих чисел від 1 до 127. Индексам променів зі стрілками відповідають значення  $i^{\wedge} > 128$  (тільки в базі НЗВ).

Множина НЗВ розбита на групи відповідно до кількості променів у вузлі та відповідає наступним значенням і діям:

а) однопроменеві вузли зустрічаються тільки на етапі виявлення стартової точки алгоритму розпізнавання БС;

б) двопроменеві НЗВ утворюють 4 підгрупи, всього з 60 зображень:

1) 12 НЗВ – апроксимують лінії (прямі) без утворення нового вузла апроксимації, якщо зміна  $\lambda$  напрямку лінії порівняно з попереднім вузлом знаходиться у діапазоні  $-\pi/k \leq \lambda < \pi/k$ ;

2) 24 НЗВ – апроксимують вигнуті лінії контурів фігур. Утворює новий вузол у разі зміни напрямку вигнутої лінії порівняно з попереднім вузлом знаходиться у діапазоні  $\pi/k \leq \lambda < 3\pi/k$ ;

3) 4 НЗВ – апроксимують кутові вузли прямокутників і прямі кути вузлів ліній зв'язків при  $\pi/2 - \pi/k \leq \lambda < \pi/2 + \pi/k$ ;

4)  $5 \times 4 = 20$  НЗВ – апроксимують усі 4 кутові вузли ромбів при  $3\pi/k \leq \lambda < \pi - 3\pi/k$ ;

в) трипроменеві НЗВ – утворюють підгрупи вузлів ліній зв'язку та відводів до прямокутника і ромба, всього  $4 \times 5 + 5 \times 4 \times 2 = 60$  зображень:

г) чотирипроменеві НЗВ – утворюють 6 зображень вузлів і один перетин ліній зв'язку.

Алгоритм генерування базового набору очевидним чином впливає з характеру представлених зображень, а саме, обертання на необхідне (4, 5, 12, 24) число раз стартових зображень навколо центру вузла. Тут тільки треба відмітити, що повинне обертатися не само зображення, а відповідним чином перераховуватися кутові параметри його генерування. Це дає можливість зберегти редактор-

ський алгоритм формування ліній і стрілок різної орієнтації. З тієї ж причини недоцільно економити на базовому наборі і обертати не НЗВ, а сканований з БС вузол.

Запропонований базовий набір НЗВ включає 131 позицію, що явно недостатньо для навчання навіть неглибокої ШНМ. Аугментація (збільшення) кількості НЗВ – метод, який дозволяє програмно отримати додаткові віртуальні навчальні приклади шляхом 2D і 3D-трансформацій візуальних форм базових зразків. У нашому випадку – це просто розширення списку програмних алгоритмів генерації і обробки зображень, а саме:

- дзеркальні відображення, повороти і зрушення в площині рисунку;

- зміна видів стрілок і товщини ліній;

- масштабування й інші довільні трансформації по горизонтальних і вертикальних осях 2D-зображення;

- зміна контрасту, яскравості кольору і фону зображень;

- додавання шуму.

Всі перераховані форми аугментації, крім першої групи, можуть бути використані для збільшення НЗВ. Частина алгоритмів автоматично підтримується в Tensorflow і PyTorch. Наприклад, масштабування і додавання шуму. Частина реалізується спеціально розробленими програмами. На рисунку показується графічне відображення результату роботи такої програми, розробленої для підготовки різноманітних можливих зображень вузлів при створенні БС.

Дзеркальні відображення не є інваріантними зображеннями одного того ж класу вузлів, так як вже представлені в базовому наборі різними класами з різними сигнатурами.

І нарешті, важливе питання для стійкості алгоритму керованого просування вікна сканера напряму зв'язано з використанням зображень, які отримані за рахунок поворотів і зсувів базових НЗВ. Такі інваріантні перетворення дійсно корисні для методів класифікації, особливо за допомогою згорткових ШНМ. Але в даному випадку завдання ШНМ не тільки класифікувати вузол, а й локалізувати (визначити) його положення в площині БС, інакше – "детектувати" вузол. При цьому достовірність успішного детектування кожного наступного вузла визначаються точністю поєднання центру

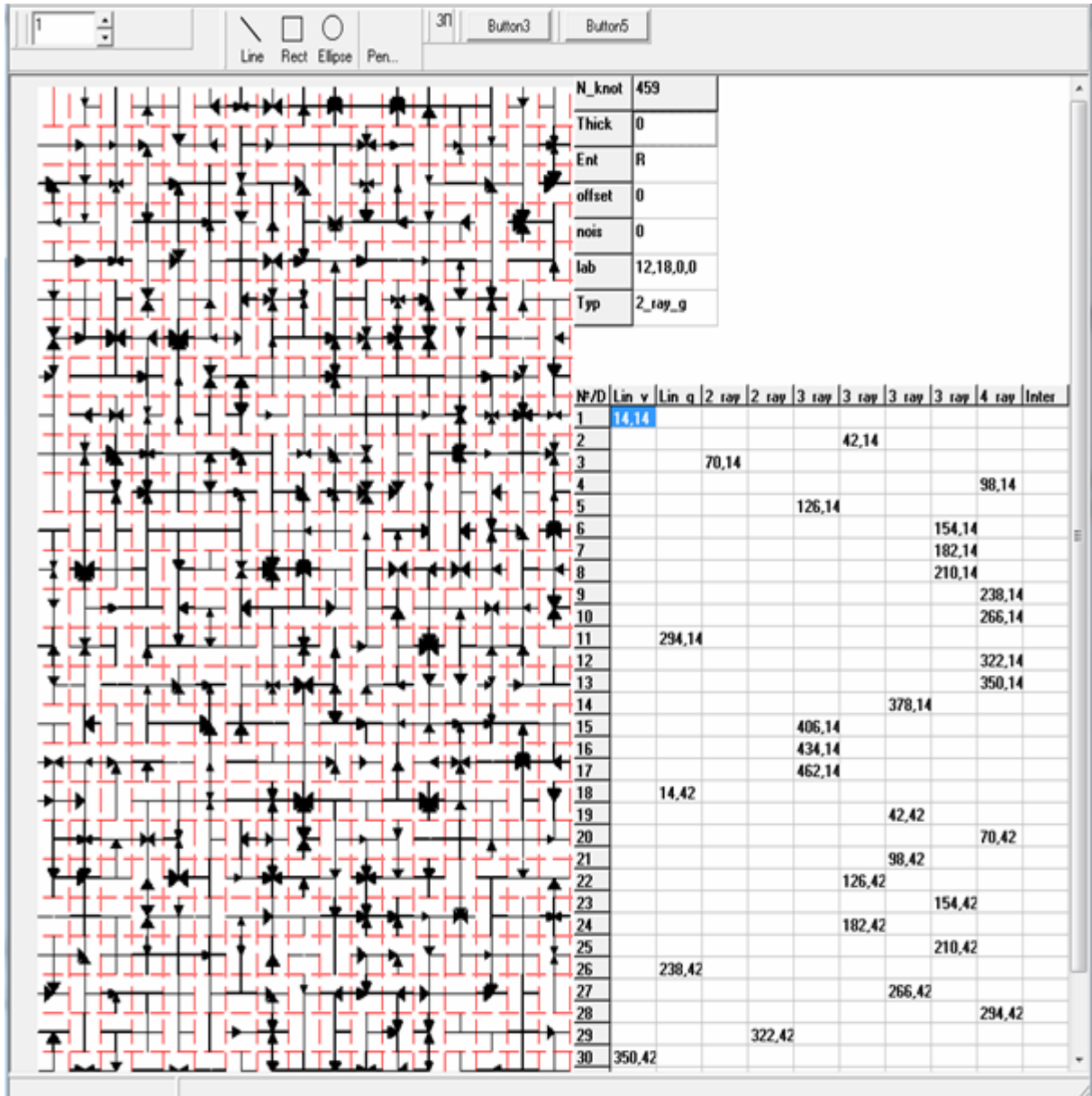


РИСУНОК. Аугментация базового набора навчальних зображень вузлів за рахунок товщини ліній і розмірів стрілок

вікна сканування з центром реального вузла. Це пов'язано з тим, що глобальні (абсолютні) координати кожного наступного вузла вздовж лінії сканування являються сумами:

$$C_{i+1}^x = C_i^x + x_i, \quad C_{i+1}^y = C_i^y + y_i,$$

де  $C_i^x, C_i^y$  – глобальні координати центру сканування поточного вузла;

$x_i, y_i$  – локальні (відносні) координати, які незалежно прописані для кожного індексу сигнатури кожного класу. Тобто положення

наступного центру визначається вектором  $V$  довжини  $v = (x_i^2 + y_i^2)^{1/2} \leq m$  (за замовчуванням  $v \approx 8$ ), проведеним з центру поточного вузла з координатами  $C_i^x, C_i^y$ , під кутом  $\tau$ , який задано індексом наступного обраного променя  $\tau = i_{t+1} \Delta \varphi$ .

**Структура моделі.** Отже помилки позиціонування центру вікна сканера за рахунок дискретної апроксимації локальних зсувів  $x_i, y_i$ , і не кратних кроку  $V$  відстаней між вуз-

лами БС, можуть накопичуватися і привести до критичних збоїв розпізнавання. Далі запропоновані варіанти організації нейромережевого моделювання БС для компенсації вказаної накопичуваної помилки. Всі методи використовують додаткове навчання ШНМ на аугментованих зображеннях вузлів зі зміщеними центрами, отриманих за рахунок зсувів  $\Delta x$  і  $\Delta y$  координат центру базових НЗВ:

$$C^x = (m-1)/2 + \Delta x, \quad C^y = (m-1)/2 + \Delta y.$$

Перший варіант нейромережевої моделі передбачає використання повністю пов'язаної ШНМ (FFN), або інакше – багатопшарового перцептрона. FFN модель має більше параметрів, чим CNN (але в розумних межах для  $m = 21$ ) і більш чутлива до зсуву зображень, тому її можна навчити розрізняти зображення з різними невеликими значеннями зсувів  $\Delta x$  і  $\Delta y$ . Для цього кожної комбінації зсувів  $\Delta x^*$  і  $\Delta y^*$  з їх загальної кількості  $\Delta x^* * \Delta y^*$  (включно базову нульову комбінацію) призначається свій клас вузлів з тією ж самою сигнатурою, але з компенсованими локальними координатами

$$x_t^* = x_t - x^*, \quad y_t^* = y_t - y^*.$$

Другий варіант – згорткова ШНМ (CNN). CNN за рахунок шарів субдискретизації більш інваріантні до масштабування і зсуву зображень, і тому краще класифікують зображення одного класу з невеликими зсувами координат  $\Delta x^*$  і  $\Delta y^*$  центру сканованих зображень вузлів. Ці зсуви поточного вузла мережею CNN ніде не фіксуються і помилкові координати наступних вузлів далі можуть накопичуватися

$$C_{t+1}^{*x} = (C_t^x + \Delta_t^*) + x_t, \quad C_{t+1}^{*y} = (C_t^y + \Delta_t^*) + y_t.$$

Усунення цього недоліку можливо шляхом виключення з CNN шарів субдискретизації. Це приведе до збільшення кількості параметрів, але знову ж таки – не набагато для ( $m = 21$ ). Навчання модифікованої CNN виконується так, як в першому варіанті.

Для навчання та дослідження запропонованих моделей ШНМ використовувався фреймворк PyTorch. Це другий за популярністю

фреймворк для побудови глибоких ШНМ, пропонує простий на базі мови Python API, підтримує тензорні динамічні обчислювальні графіки, автоматичне диференціювання і розрахунок градієнтів, у тому числі з використанням GPU. Спадкування функції базового класу nn.Module дозволяє створювати і ефективно використовувати практично всі архітектури і технології глибоких ШНМ.

**Післямова.** Повертаючись до зауваження про необхідності 100 % покриття варіантів зв'язків БС, слід все ж уточнити, що людина цілеспрямовано мінімізує довжину та спрощує конфігурації зв'язків між фігурами БС. Системи автоматизованого проектування роблять це ще краще. Але можливість появи в БС навіть одного заплутаного зв'язку потребує навчати ШНМ на всіх можливих конфігураціях таких зв'язків, а це теж нереально.

З іншого боку, необхідність подання структур, процесів і алгоритмів у вигляді БС виникає в разі складності побудови однозначного для сприйняття відповідного текстового опису. Тобто в БС з самого початку закладена більш проста і формалізована мова уявлень, ніж природна мова людини, і тому складність розпізнавання таких конструкцій має бути технічно простіше. Досягнення в області обробки (наприклад, переведення) природних мов за останні роки дуже вражаючі. В першу чергу це стало можливо за рахунок застосування таких нейромережевих технологій, як рекурентне перетворення (трансляція) текстової послідовності в іншу послідовність – seq2seq [3], векторне семантичне представлення (embedding) слів в word2vec [4], облік елементів уваги – attention [5]. Тому уявити як одну із сторін такої трансляції довільне зображення, в тому числі БС, цілком можливо, що і було зроблено [6].

Основна складність тут полягає у тому, що ШНМ все ще не спроможні самостійно гарантувати достатню точність детектування всіх складових зображення і трансляції сукупної композиції в адекватні семантичні (особливо технічні) характеристики.

Людина розпізнає зв'язки на зображеннях вибірково їх скануванням на основі, наприклад, уваги, спільним розв'язуванням за допомогою логічних (синтаксичних) або образних (нейромережевих) підходів. При цьому в ієрархії об'єктів, що утворюють зображення, для кожної системи розпізнавання, вочевидь, існує

деяке оптимальне співвідношення величин складності самих об'єктів і зв'язків між ними. Наприклад, людина розпізнає заплутані зв'язки в БС послідовним скануванням їх первинних графічних примітивів, в якості яких, в кращому разі, виступають закінчені відрізки ліній між вузлами, візуально відмітними від самої лінії. Скоріш за все, це також буде найкращий автоматично обраний рівень деталізації зв'язків БС у складі наступних нейромережево-синтаксичних (або нейрологічних) засобів розпізнавання і опису довільних зображень і відео.

**Висновки.** Розроблена нейромережева модель розпізнавання графічних примітивів у складі засобів синтаксичного аналізу графічних зображень блок-схем. Сформована базова множина навчальних і тестових зображень вузлів БС. Розглянуто етапи програмного синтезу й аугментації синтетичних навчальних даних. Запропоновано нейромережеві моделі для підвищення точності процесів детектування вузлів БС на основі повнозв'язаних та згорткових мереж. Обґрунтовано доцільність поєднання синтаксичних та нейромережевих методів розпізнавання для широкого класу графічних зображень.

#### СПИСОК ЛІТЕРАТУРИ

1. Чічирін Є.М. Распознавание структуры графических изображений блок-схем.. *Комп'ютерні засоби, мережі та системи*. 2017. № 16. С. 87 – 96.

2. LeCun, Yann. LeNet-5, convolutional neural networks. November 2013.
3. Sutskever I., Vinyals O., & Le Q.V. Sequence to sequence learning with neural networks. *In Advances in Neural Information Processing Systems*. 2014.
4. Mikolov T., Corrado G., Chen K., & Dean J. Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations*. ICLR 2013.
5. Bahdanau D., Cho K., & Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate. *In ICLR 2015*.
6. Vinyals O., Toshev A., Bengio S., & Erhan, D. Show and tell: A neural image caption generator. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015. P. 3156–3164.

#### REFERENCES

1. Chichirin E.N. Recognition of the structure of graphic images of flowcharts. *Computer system, measure and system*. 2017. No. 16. S. 87 – 96.
2. LeCun, Yann. LeNet-5, convolutional neural networks. November 2013.
3. Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *In Advances in Neural Information Processing Systems*.
4. Mikolov, T., Corrado, G., Chen, K., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations (ICLR 2013)*.
5. Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. *In ICLR 2015*.
6. Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. *In Proceedings of the IEEE conference on computer vision and pattern recognition (P. 3156–3164)*.

Одержано 24.09.2019