



Р. І. Кордюк¹, О. В. Овсяк²

¹ Національний університет "Львівська політехніка", м. Львів, Україна

² Національний лісотехнічний університет України, м. Львів, Україна

ДЕРЕВОПОДІБНА МОДЕЛЬ КОЛАЖІВ З ПОПЕРЕЧНИМИ ЗВ'ЯЗКАМИ

Наведено інтерактивну деревоподібну модель колажів із поперечними зв'язками, розроблену для ефективного подолання обмежень, притаманних наявним редакторам графічних зображень, таким як Photoshop Online, Canva та Pixlr Editor. На підставі проведеного детального аналізу наявних популярних рішень встановлено, що основними недоліками цих так званих редакторів колажів є відсутність автоматичних зв'язків між контейнерами елементів зображення та потреба у значних ручних налаштуваннях структури колажу під час редагування відповідних композицій. Унаслідок цього виникають проблеми у дотриманні візуальної узгодженості та гармонійності за зміни розмірів зображення або позицій окремих елементів структури колажу. Розроблену модель колажу побудовано як чітко структуроване ієрархічне дерево з чотирма рівнями вкладеності: кореневий елемент, рядки, стовпці та контейнери для елементів зображення. Істотним покращенням є впровадження поперечних зв'язків між сусідніми контейнерами, які забезпечують автоматичний двосторонній перерахунок позицій та розмірів контейнерів зображення в разі зміни параметрів одного з елементів. Детально описано математичні основи цієї моделі, зокрема застосування принципів теорії графів і дерев, що дає змогу формально визначити логіку взаємодії елементів у колажі. Реалізацію моделі виконано засобами мови програмування TypeScript та бібліотеки Fabric.js, з активним використанням патернів проектування Composite для структурування компонентів та Observer для ефективного управління реактивністю та змінами стану системи у режимі реального часу. Проведено експериментальні дослідження роботи створеної моделі, які показали істотне покращення процесу редагування графічних композицій: скорочується кількість потрібних дій користувача у декілька разів порівняно із традиційними моделями. Запропонована система демонструє високу продуктивність, швидкість реагування на зміни та стабільність підтримки візуальної гармонії композиції колажу навіть за умов інтенсивного редагування елементів. Виявлено, що розроблена модель колажу успішно поєднує переваги автоматизованих інтелектуальних рішень та зберігає гнучкість і точність ручних редагувань, що робить її ефективним інструментом для широкої аудиторії користувачів.

Ключові слова: площинний колаж; деревоподібна структура; графічний редактор колажів; алгоритм handleMoving; композиція зображення.

Вступ / Introduction

Колажі є популярним способом наведення групи елементів зображення в одній площині, широко використовуваним у соціальних мережах, рекламі та цифровому дизайні. Останніми роками характерний активний розвиток алгоритмів автоматизованого створення колажів, що має на меті спростити і пришвидшити процес компонування відповідних зображень. Наявні дослідження можна умовно поділити на кілька напрямів: автоматичне компонування зображення на підставі моделей покращення розміщення його елементів; моделі врахування семантичної взаємозв'язності елементів зображення, а також застосування глибинного навчання для покращення самої композиції зображення.

Однією з перших відомих систем автоматичного створення колажів є AutoCollage [16]. Система AutoCollage застосовує графовий алгоритм для визначення областей інтересу на фотографіях і безшовно об'єднує їх у фінальний колаж за допомогою згладжування країв. Поп-

ри відчутну ефективність у створенні узгоджених композицій, ця система має істотні недоліки – збереження зайвого фону та відсутність аналізу семантичних зв'язків між елементами зображення. Тобто система AutoCollage не гарантує, що тематично чи змістовно пов'язані знімки розташуються поруч, оскільки не виконується семантичне групування елементів зображення.

Для подолання цих обмежень розробники аналогічних систем запропонували низку їх удосконалень. Зокрема, система Correlation-Preserving Photo Collage [11] інтегрує глибокі нейронні мережі для визначення змістовної кореляції між елементами зображення. Це дає змогу автоматично групувати семантично споріднені зображення разом, роблячи їхню композицію більш інформативною та осмисленою. Внаслідок роботи такої системи, колажі сприймаються набагато краще: пов'язані за змістом елементи зображення розміщуються близько одне до одного, а простір площини використовується значно компактніше.

Інформація про авторів:

Кордюк Роман Ігорович, аспірант, кафедра комп'ютерних технологій у видавничо-поліграфічних процесах.

Email: roman.i.kordiuk@lpnu.ua; <https://orcid.org/0009-0008-6082-901X>

Овсяк Олександр Володимирович, д-р техн. наук, професор, кафедра комп'ютерних наук. Email: ovsjak@nltu.edu.ua;

<https://orcid.org/0000-0003-2620-1938>

Цитування за ДСТУ: Кордюк Р. І., Овсяк О. В. Деревоподібна модель колажів з поперечними зв'язками. Науковий вісник НЛТУ України. 2025, т. 35, № 4. С. 123–133.

Citation APA: Kordiuk, R. I., & Ovsyak, O. V. (2025). Tree-based collage model with transverse-connections. *Scientific Bulletin of UNFU*, 35(4), 123–133. <https://doi.org/10.36930/40350414>

Багато інших сучасних моделей автоматичного створення колажів застосовують метод глибокого навчання з підкріпленням. Зокрема, в роботі [20] розроблено модель компонування колажів із застосуванням глибокого навчання з підкріпленням, що вчиться розміщувати елементи зображення послідовно, отримуючи винагороду за естетичні показники композиції колажу – компактність, правильні пропорції, відсутність перекриттів тощо. Така модель значно підвищила естетичну якість та ефективність автоматичного створення колажів, наблизивши композицію до рівня ручної роботи кваліфікованого дизайнера. Окрім цього, у 2022 р. запропоновано диференційовану модель SoftCollage [19], що автоматично генерує колаж через ймовірнісне деревоподібне подання макету з можливістю навчання мережі на підставі градієнтів. Модель SoftCollage унікає спотворення пропорцій елементів зображення та досягає високої швидкодії завдяки диференційованій структурі дерева; проведені експерименти показали здатність цієї моделі створювати компактні та семантично узгоджені колажі.

Попри успіхи автоматизованих моделей створення колажів, вони переважно зосереджені на побудові композиції колажу, а не на її інтерактивному редагуванні користувачем. У практиці ж дизайнерських рішень часто виникає потреба вручну налаштовувати макет композиції колажу: змінювати розміри або положення окремих частин колажу після початкового компонування. Популярні онлайн-інструменти для створення колажів – такі як Photoshop Online [1], Canva [3], Pixlr Editor (Online Photo Editor) [15] – надають користувачам базові можливості компонування, але мають обмеження у гнучкості редагування. Наприклад, модель колажу в середовищі Photoshop Online побудовано на фіксованій прямокутній сітці: користувач задає кількість рядків і стовпців, утворюючи рівномірну решітку, у комірки якої вставляють елементи зображення. Контейнери із фото в такій моделі існують незалежно один від іншого, тому її основним недоліком є відсутність зв'язку між контейнерами: зміна розміру одного блоку ніяк не позначається на сусідніх блоках. Будь-яке перепланування колажу потребує ручного редагування решти його елементів, що є трудомістким процесом і, як наслідок, знижує точність їх розташування. Аналогічно, сервіс Canva пропонує набір заздалегідь визначених шаблонів можливих макетів колажу; після вибору шаблону користувач розміщує зображення у фіксованих рамках. Хоча це спрощує початкове створення колажу, подальша зміна його структури (наприклад, поділ однієї рамки на дві або об'єднання двох областей) зазвичай не підтримується безпосередньо сервісом Canva.

Отже, існує прогалина між автоматичними алгоритмами побудови колажів та інструментами інтерактивного їх редагування. Автоматизовані моделі покращення розташування елементів намагаються одразу надати фінальний результат, не залишаючи місця для гнучкого налаштування користувачем, тоді як наявні редактори пропонують тільки найпростіші схеми компонування без внутрішньої "логіки" зв'язків між елементами. Це мотивує процес розроблення моделі, що поєднує переваги обох: забезпечує структурованість моделі колажу, в якій елементи пов'язані між собою і, водночас, дає змогу користувачу легко змінювати композицію, не порушуючи її цілісності.

Об'єкт дослідження – побудова моделі колажів з поперечними зв'язками.

Предмет дослідження – методи і засоби побудови деревоподібної моделі колажів з поперечними зв'язками, яка автоматизує підтримку цілісності композицій у процесі їх редагування.

Мета роботи – розробити деревоподібну модель колажів з поперечними зв'язками між їхніми елементами, яка дасть змогу інтегрувати автоматичні алгоритми їх компонування з інтерактивними процедурами редагування, а також підвищить зручність управління отриманими композиціями завдяки структурній узгодженості та взаємозалежності їхніх елементів.

Для досягнення зазначеної мети визначено такі основні завдання дослідження:

- проаналізувати сучасні моделі автоматизації роботи з колажами, що дасть змогу визначити їхні переваги й недоліки, а також сформулювати вимоги до створення більш адаптивних та зручних для користувачів редакторів колажів;
- розробити концептуальну чотирирівневу модель подання колажу, що дасть змогу інтегрувати автоматичні алгоритми їх компонування з інтерактивними процедурами редагування та підвищить зручність управління отриманими композиціями завдяки структурній узгодженості та взаємозалежності їхніх елементів;
- розробити алгоритми зміни структури колажу (переміщення меж між елементами, додавання/видалення елементів) з автоматичним оновленням суміжних областей і реалізувати ці алгоритми у вигляді функцій редактора колажів та оцінити їх ефективність (складність реалізації, зручність використання, переваги над наявними рішеннями), що забезпечить практичну цінність розробленої системи у завданнях створення графічних композицій, зменшить час на редагування макетів колажу і підвищить якість отриманого результату.

Аналіз останніх досліджень та публікацій. Системи автоматичного компонування колажів, наведені в літературі, демонструють різні концепції досягнення візуально привабливого результату. Розглянемо кілька ключових робіт і порівняємо їх з нашою моделлю.

У дослідженні [16] автори створили систему AutoCollage, яка автоматично формує колажі з фотографій без використання фіксованих шаблонів. Автори проаналізували алгоритми комп'ютерного зору для вибору найбільш цікавих областей зображень та комбінаторного впорядкування їх розташування. Встановлено, що хоча система AutoCollage мінімізує ручну працю користувача, вона не враховує семантики фотографій і може містити зайві деталі.

У роботі [11] наведено модель Correlation-Preserving Photo Collage, де автори дослідили можливості нейронних мереж для семантичного групування елементів зображення перед компонуванням колажу. Підхід дає змогу генерувати колажі з тематично близькими фото, розміщеними поряд, використовуючи метод нерегулярного розбиття області колажу для ефективного заповнення простору.

У дослідженні [20] запропоновано модель компонування колажів з використанням глибокого навчання з підкріпленням DRL (англ. *Deep Reinforcement Learning*). Автори дослідили процес поступового додавання фото на полотно, де модель колажу навчається естетичним критеріям на підставі великих наборів даних і оцінок користувачів, забезпечуючи високі показники естетичності отриманих композицій.

У роботі [19] автори створили диференційовану модель SoftCollage, яка генерує бінарне дерево поділів області колажу за допомогою нейронних мереж. Дослідження показало переваги такої деревоподібної моделі у забезпеченні структурної цілісності, відсутності перекриттів та гнучкості компоновання.

У дослідженні [22] автори запропонували генеративно-змагальну мережу EcGAN, яка створює графічні макети з фіксованою кількістю та типами елементів. Запропонована модель генерування графічних колажів використовує обмежувальні рамки, маски елементів зображення та дворівневі дискримінатори, що дає змогу створювати композиції зображень з урахуванням специфічних умов і є корисною для графічного дизайну.

У дослідженні [6] автори описали розроблену систему Iris, орієнтовану на користувача, яка забезпечує інтерактивне середовище для генерування графічних макетів колажу з урахуванням множинних обмежень. Дизайнер може вказати певні обмеження, такі як вирівнювання елементів зображення або залишення певних областей порожніми, після чого генеративна модель системи пропонує відповідні макети. Ця система вказує на важливість контролю та інтерактивності за практичних умов, даючи можливість дизайнерам швидко отримувати якісні пропозиції, які відповідають їхнім вимогам.

Проведений аналіз останніх досліджень та публікацій засвідчив значний прогрес у напрямі автоматичного та напівавтоматичного компоновання колажів, насамперед щодо застосування глибокого навчання, удосконалення макетів колажу та інтеграції семантичних зв'язків між елементами зображення. Однак жодна з відомих моделей не забезпечує поєднання чотирирівневої деревоподібної структури зображення з динамічними поперечними зв'язками його елементів та можливістю повноцінного інтерактивного редагування й автоматичного оновлення композиції колажу у реальному часі. Отже, виникає потреба у розробленні нового редактора фотографій, який дає змогу об'єднати переваги наявних рі-

шень і водночас усунути їхні головні недоліки, забезпечуючи більш гнучке та інтуїтивне оброблення площинних колажів.

Матеріали та методи дослідження. Моделі колажів та алгоритми їхнього оброблення засобами інформаційних технологій і систем. Метод теорії графів застосовано для створення математичної моделі колажу у вигляді деревоподібної структури колажу з поперечними зв'язками його елементів. Для розроблення моделей колажів та їх оброблення застосовано теорію алгоритмів.

Практичну реалізацію створених моделей виконано у спосіб застосування фреймворку Angular, мов програмування TypeScript і HTML, Canvas технології через бібліотеки Fabric.js, функції алгоритму handleMoving, інтегрованого в обробники подій повзунків з активним використанням патернів проектування Composite для структурування компонентів та Observer для ефективного управління реактивністю та змінами стану системи у режимі реального часу.

Результати дослідження та їх обговорення / Research results and their discussion

Унаслідок проведеного дослідження створено чотирирівневу деревоподібну модель колажу з поперечними зв'язками його елементів, яка забезпечує інтерактивне редагування графічних композицій зі збереженням цілісності макету в реальному часі. Запропонована модель колажу базується на графовій структурі, що дає змогу автоматично підтримувати узгодженість між її елементами та спрощує процес компоновання та коригування колажу. Далі наведено детальний опис розробленої моделі колажу, її структури та функціональних особливостей.

На рис. 1 наведено створену чотирирівневу деревоподібну модель колажу з поперечними зв'язками його елементів у вигляді графа.

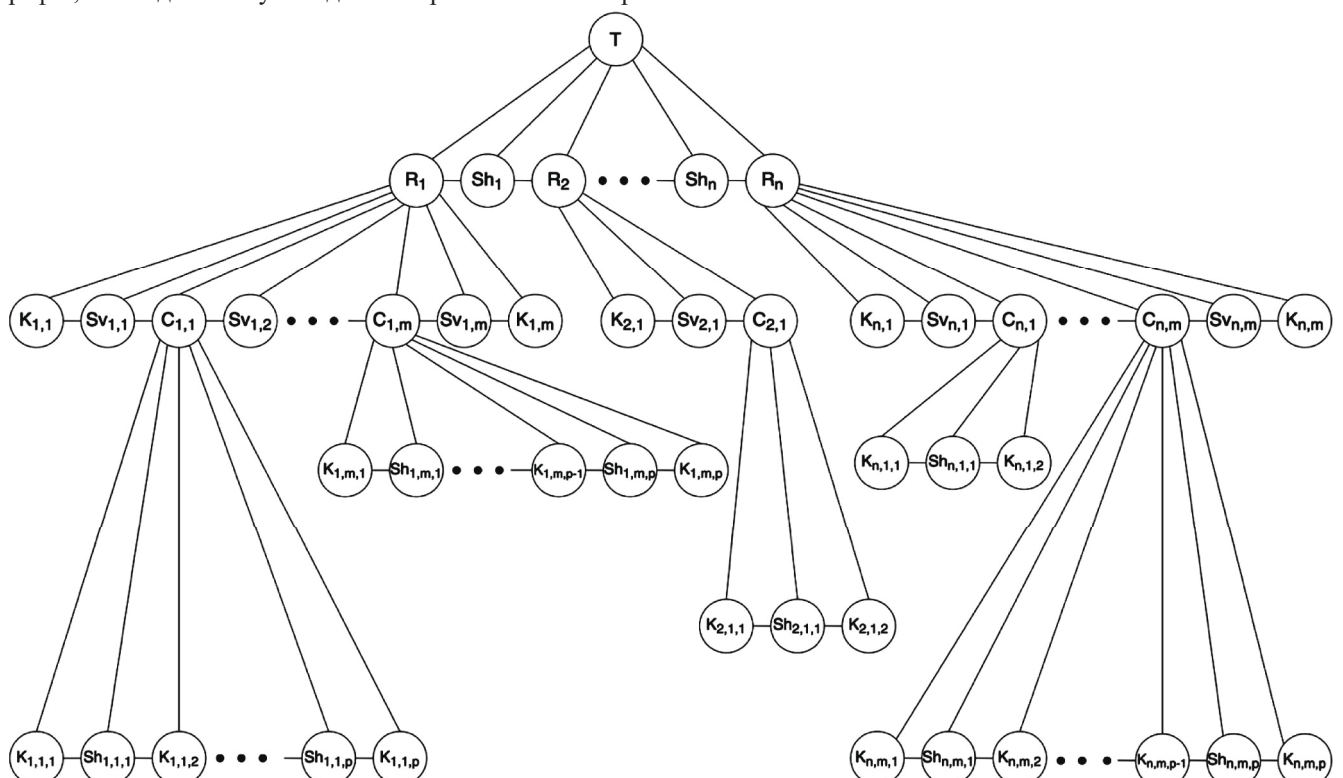


Рис. 1. Загальний вигляд чотирирівневого графа деревоподібної моделі колажів / General view of the four-level graph of a tree-based collage model

Наведена на рис. 1 деревоподібна модель колажу має чотири рівні вкладеності ієрархічної структури та розрахована на інтерактивну організацію композиції графічних зображень із автоматизованими зв'язками елементів композиції та можливістю редагування структури колажу).

На *першому* рівні розташований єдиний кореневий вузол (Т), що є базовою опорною точкою всієї моделі. Корінь визначає загальні межі композиції колажу та слугує головним вузлом, від якого розгалужуються всі інші структурні елементи.

Другий рівень містить сукупність n горизонтальних структурних елементів, які називаються рядками (позначають як $R_1, R_2, \dots, R_n$). Рядки організовано горизонтально та відділені один від іншого спеціальними вертикальними регуляторами (горизонтальними повзунками Sh_i). Ці повзунки дають змогу користувачеві змінювати висоту одного рядка з автоматичним перерахунком і пропорційною адаптацією сусідніх рядків. У такий спосіб забезпечується стабільність і гармонія композиції колажу за вертикальних її трансформацій.

На *третьому* рівні від кожного рядка відходять m вертикальних структурних елементів – стовпців ($C_{i,1}, C_{i,2}, \dots, C_{i,m}$), де i – номер рядка. Окрім стовпців, на цьому ж рівні розташовані контейнери елементів зображення ($K_{i,j}$) та вертикальні повзунки ($Sv_{i,j}$). Вертикальні повзунки переміщуються по горизонталі, забезпечуючи можливість змінювати ширину контейнерів елементів зображення і стовпців із автоматичним коригуванням сусідніх елементів, що забезпечує рівномірність горизонтального розподілу їхнього вмісту.

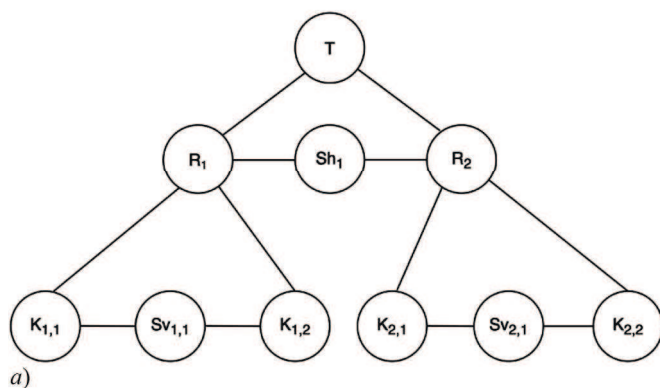


Рис. 2. Приклад трирівневої моделі колажу (а) і її можливе наповнення (b) / Example of the three-level collage model (a) and its possible content (b)

1. Перший рівень (корінь) – кореневий елемент графа (Т) становить загальний контейнер колажу, який охоплює всі нижчі рівні елементів зображення і визначає загальний простір композиції колажу.
2. Другий рівень (рядки та горизонтальний повзунок) – містить два рядки, позначені як R_1 та R_2 . Вони відповідають верхній та нижній горизонтальним зонам композиції колажу. Між цими рядками розташовано горизонтальний повзунок (позначений Sh_1), який дає змогу змінювати вертикальне співвідношення їх висоти, тобто переміщувати межу між рядками вгору або вниз.
3. Третій рівень (контейнери та вертикальні повзунки) – кожен рядок має два контейнери для елементів зображення:
 - верхній рядок (R_1) містить контейнери елементів зображення $K_{1,1}$ та $K_{1,2}$, розділені вертикальним повзунком ($Sv_{1,1}$), який відповідає за горизонтальне регулювання ширини цих контейнерів;

На *четвертому*, найнижчому рівні, розташовані додатково вкладені структури: p контейнерів елементів зображення ($K_{i,j,1}, K_{i,j,2}, \dots, K_{i,j,p}$), де i та j – відповідно номери рядка та стовпця. Ці контейнери відповідають за безпосереднє розміщення елементів зображень і деталізують композицію на нижньому рівні. Між цими контейнерами також розташовані повзунки ($Sh_{i,j,p}$), які дають змогу здійснювати регулювання висоти, при цьому автоматично перераховуючи параметри сусідніх елементів.

Особливістю моделі колажу є поперечні зв'язки між вузлами всередині кожного рівня (рядками, стовпцями, контейнерами). Ці зв'язки реалізують механізм автоматичного двостороннього перерахунку параметрів елементів зображення за зміни одного з них, що ґрунтується на алгоритмі handleMoving. У такий спосіб забезпечується автоматична адаптивність структури колажу до змін, підтримується цілісність та узгодженість композиції колажу без додаткових ручних налаштувань.

Ця модель колажу дає змогу ефективно керувати складними багаторівневими графічними композиціями, істотно спрощуючи процес редагування та забезпечуючи високу гнучкість та точність у налаштуванні параметрів утворених зображень.

На рис. 2 наведено приклад трирівневої моделі колажу (а) і її можливе наповнення (b). На поданих зображеннях показано трирівневу деревоподібну модель колажу, яку наведено у вигляді графа і яка відображає композицію колажу з чотирьох контейнерів елементів зображення, організованих у два рядки та розмежованих повзунками для динамічного регулювання розмірів зображення.

- нижній рядок (R_2) містить контейнери елементів зображення $K_{2,1}$ та $K_{2,2}$, розділені аналогічним вертикальним повзунком ($Sv_{2,1}$), що виконує ту ж функцію горизонтального переміщення меж між контейнерами.

На рис. 3 подано приклад графа чотирирівневого колажу (а) і його можливе наповнення (b). Наведений граф описує чотирирівневу деревоподібну модель колажу, що відповідає структурі зображення, а саме:

- 1) перший рівень (корінь) – кореневий елемент графа (Т). Він є початковим вузлом всієї структури колажу, який об'єднує всі нижче розташовані елементи;
- 2) другий рівень (рядки та горизонтальний повзунок):
 - елементи R_1 та R_2 відповідають за два рядки колажу;
 - між рядками розташований горизонтальний повзунок Sh_1 , що дає змогу змінювати вертикальну висоту рядків та відповідно пропорції зображень у цих рядках;
- 3) третій рівень (контейнери, стовпці та вертикальні повзунки):

- у першому рядку (R_1): контейнери для елементів зображення $K_{1,1}$, $K_{1,2}$; стовпець $C_{1,1}$, який містить додаткові елементи зображення; два вертикальні повзунки $Sv_{1,1}$ та $Sv_{1,2}$, які дають змогу змінювати горизонтальну ширину контейнерів і стовпців;
- у другому рядку (R_2): контейнери для елементів зображення $K_{2,1}$, $K_{2,2}$; стовпець $C_{2,1}$, який містить додаткові елементи зображення; два вертикальні повзунки $Sv_{2,1}$ та $Sv_{2,2}$, аналогічні повзункам першого рядка, що відповіда-

- ють за горизонтальні зміни ширини відповідних контейнерів та стовпців;
- 4) четвертий рівень (елементи всередині стовпців і їхні повзунки):
 - у стовпці $C_{1,1}$: два контейнери для елементів зображення $K_{1,1,1}$ і $K_{1,1,2}$; один горизонтальний повзунок $Sh_{1,1,1}$, що дає змогу змінювати вертикальну висоту цих контейнерів;
 - у стовпці $C_{2,1}$: два контейнери для елементів зображення $K_{2,1,1}$ і $K_{2,1,2}$; один горизонтальний повзунок $Sh_{2,1,1}$, що аналогічно регулює вертикальну висоту цих контейнерів.

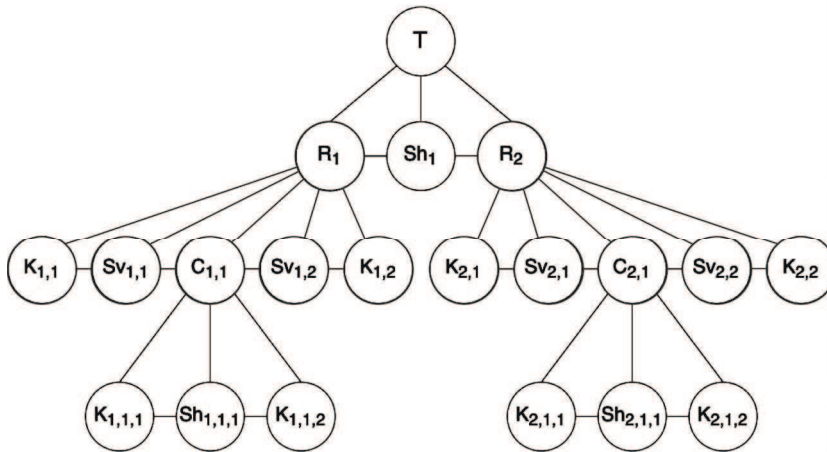


Рис. 3. Приклад графа чотирирівневого колажу (a) і його можливе наповнення (b) / Example of a four-level collage graph (a) and its possible content (b)



Рис. 4. Приклад руху повзунка Sh_1 вертикально донизу у чотирирівневному колажі (a) та повзунка $Sv_{1,2}$ у R_1 спочатку горизонтально ліворуч потім праворуч (b) / Example of moving a slider Sh_1 vertically downward in a four-level collage (a), and moving the slider $Sv_{1,2}$ in R_1 first horizontally to the left, then to the right (b)



Рис. 5. Приклад руху повзунка $Sh_{1,1,1}$ вертикально донизу у стовпці $C_{1,1}$ (a) та руху повзунка $Sh_{2,1,1}$ вертикально догори у стовпці $C_{2,1}$ (b) / Example of moving a slider $Sh_{1,1,1}$ vertically downward in column $C_{1,1}$ (a) and moving a slider $Sh_{2,1,1}$ vertically upward in column $C_{2,1}$ (b)



Рис. 6. Приклад руху повзунка Sh_1 вертикально догори (a), та донизу до можливих мінімальних меж (b) / Example of moving a slider Sh_1 vertically upward (a), and downward to the possible minimum boundaries (b)

Моделі автоматичного оновлення (handleMoving) структури та елементів колажу. Основна функціональність, яка забезпечує поперечні зв'язки, зосереджена в методах класів handleMoving для різних рівнів моделі колажу. Розглянемо алгоритм для CollageTree та повзунка, що рухається вертикально, змінюючи висоти двох сусідніх рядків колажу (метод CollageTree.handleMoving). Нехай користувач перетягує повзунок, що розташований між двома рядками – верхнім (попереднім) і нижнім (наступним). За кожного кроку руху обчислюється величина зсуву Δy (у пікселях) – наскільки повзунок змістився від початкової позиції

$$\Delta y = y - y', \quad (1)$$

де: y' – початкова ордината повзунка; y – теперішня ордината повзунка. Далі алгоритм виконує такі кроки:

Перевірка обмежень розміру елементів колажу. Перед тим як змінювати висоти рядків, алгоритм перевіряє, чи не зробить запланований зсув якийсь рядок надто малим. Наприклад, ми встановили мінімальну висоту рядка 50 пікселів: якщо рух повзунка призведе до того, що висота одного з рядків стане менше цього мінімуму, то рух потрібно обмежити. Для перевірки аналізуються всі зображення, вкладені в потенційно проблемний рядок (зокрема, якщо рядок містить стовпець, перевіряються всі зображення у ньому). Якщо хоча б один стовпець чи рядок вже на межі мінімуму, прапорець canMoveFarther отримує значення false, і повзунок повертається на стару позицію. У такий спосіб гарантується збереження мінімально допустимих розмірів всіх частин колажу.

Пропорційне оновлення розмірів сусідніх рядків. Якщо обмеження не порушені (тобто canMoveFarther лишився true), обчислюються нові висоти двох суміжних рядків. Верхній рядок збільшується або зменшується на величину y , а нижній – на протилежну величину (зменшується або збільшується на Δy). Формально, якщо H_{prev} і H_{next} – початкові висоти рядків, після переміщення повзунка на Δy отримаємо:

$$H_{prev} = H_{prev} + \Delta y, H_{next} = H_{next} - \Delta y, \quad (2)$$

за умови, що Δy додатне за зміщення повзунка вниз (збільшуючи верхній рядок та зменшуючи нижній) і від'ємне – за зміщення вгору. Одночасно з висотами зміщуються і вертикальні координати рядків: верхній рядок зсувається на $\Delta y/2$ вниз (щоб залишатися суміжним з повзунком), а нижній – на $\Delta y/2$ вгору. У реалізації

Fabric.js достатньо встановити нові властивості *height* і *top* для об'єктів рядків, при цьому метод *setCoords()* оновить координати.

Оновлення вкладених елементів зображення (стовпців). Якщо рядки містять стовпці (CollageColumn), то зміна їх висоти вимагає пропорційної зміни висот внутрішніх елементів зображення у цих стовпцях. Алгоритм проходить по кожному стовпцю у змінених рядках і масштабує її вміст: збільшує або зменшує висоти вкладених елементів зображення так, щоб сумарна висота заповнила новий розмір стовпця, зберігаючи пропорції між елементами зображення. Граничні випадки, коли якісь елементи зображення вже мали мінімальну висоту, було враховано на етапі 2 і блокують подальше масштабування такого стовпця. У коді для цього застосовано допоміжні методи на зразок *handleColumnBlockMove* та *handleColumnReverseBlockMove* (для випадків руху вниз чи вгору), що ділять додану/відняту висоту рівномірно між дочірніми елементами стовпця.

Блок-схему алгоритму *handleMoving* для *CollageTree* наведено на рис. 7. Аналогічні алгоритми реалізовано для вертикальних повзунків у межах рядка (метод *CollageRow.handleMoving*). Коли користувач перетягує вертикальний повзунок між стовпцями або зображеннями в рядку, обчислюється зсув Δx по осі абсцис. Далі визначаються сусідні елементи – лівий (*previousEl*) і правий (*nextEl*). Якщо обидва сусіди є, алгоритм перевіряє типи цих елементів. У разі, коли праворуч або ліворуч розташований стовпець, змінювати потрібно ширину кожного вкладеного зображення в цьому стовпці. У нашій реалізації:

- для правого сусіда (*nextEl*) типу *CollageColumn*: всі його дочірні зображення звужуються або розширюються на Δx , зсув їх координати X на $\Delta x/2$, і сам стовпець також змінює ширину на $-\Delta x$. Якщо правий сусід – одиночний контейнер зображення, його ширина просто змінюється на $-\Delta x$, координата зсувається на $\Delta x/2$;
- для лівого сусіда (*previousEl*) типу *CollageColumn*: аналогічно, всі його дочірні зображення розширюються на Δx , і стовпець збільшує ширину на $+\Delta x$. Якщо ліворуч звичайний контейнер, йому додається Δx до ширини.

Після цього використовують метод *updateChildren()* для перерахунку позицій всіх дочірніх елементів рядка, згідно з оновленими розмірами, та оновлюються координати контейнера (*setCoords*). Canvas перемальовується методом *requestRenderAll()* для відображення змін.

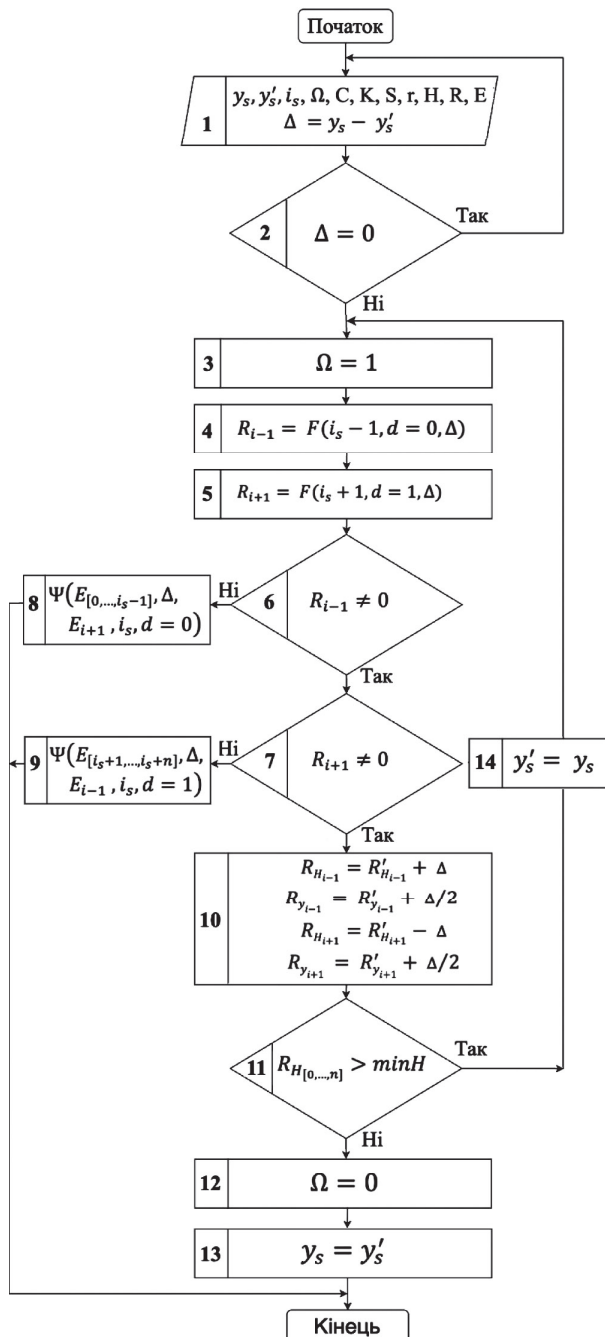


Рис. 7. Блок-схема алгоритму handleMoving для CollageTree / Flowchart of the handleMoving algorithm for CollageTree

Окремо реалізовано методи для видалення елементів, додавання нового рядка чи стовпця. Наприклад, у разі видалення елемента зображення із стовпця перевіряється, чи треба також видалити відповідний повзунок, та перерозподіляється висота, раніше зайнята цим елементом зображення, між рештою елементів стовпця. Подібні операції коректно підтримують цілісність структури колажу: після видалення фото сусідні елементи автоматично змінюють розмір, щоб заповнити порожній простір. За додавання елемента зображення у стовпець перевіряється, чи треба також додати відповідний повзунок, та перерозподіляється висота, виділяється висота для даного зображення, а усі наявні елементи пропорційно зменшують свою висоту

$$H_{prev} = H_{prev} + \Delta y, H_{next} = H_{next} - \Delta y. \quad (3)$$

Програмна реалізація структури колажу і шаблони проєктування. Модель колажу реалізовано у вигляді об'єктно-орієнтованої системи класів мовою програмування

TypeScript із використанням бібліотеки FabricJS (для роботи з HTML5 Canvas). Кожен структурний елемент моделі колажу відповідає окремому класу:

1. CollageTree – клас кореневого об'єкта колажу (успадковує базовий клас fabric.Object).
2. CollageRow – клас рядка колажу.
3. CollageColumn – клас стовпця в рядку.
4. CollageSeparator – клас повзунка (успадковує fabric.Rect, оскільки реалізується у вигляді тонкого прямокутника, що реагує на події перетягування).
5. @types/fabric – офіційний пакет TypeScript-типів для бібліотеки Fabric.js

Діаграму класів редактора колажів наведено на рис. 8.

Для полегшення інтеграції з бібліотекою FabricJS використано патерн "Реєстратор": кожен клас реєструється у глобальному об'єкті fabric (наприклад, fabric.CollageRow = fabric.util.createClass(...)), що дає змогу створювати його екземпляри засобами бібліотеки FabricJS. Взаємозв'язок між класами відображає ієрархію моделі колажу: клас CollageTree містить масив дочірніх об'єктів children (рядків та повзунків), CollageRow – масив своїх children (стовпців, контейнерів елементів зображення та повзунків), CollageColumn – масив children (контейнерів елементів зображення чи повзунків). Отже, застосовано патерн Компонувальник (Composite): колаж і його складові наведено однорідно як об'єкти з методом отримання вкладених елементів (getChildren()), що дає змогу рекурсивно обходити все дерево і виконувати операції над кожним елементом однаковим способом. Наприклад, метод встановлення координат setCoords() реалізований так, що спочатку викликається рекурсивно для всіх вкладених елементів, а потім для батьківського контейнера, що гарантує оновлення всіх меж елементів зображення після змін їх розмірів.

Ще один важливий аспект – подійно-орієнтована взаємодія. Повзунок (CollageSeparator) відстежує події користувацького перетягування (dragging). На початку перетягування повзунка (mousedown на ньому) фіксується його початкова позиція (властивості oldX або oldY). Під час руху (moving подія) повзунок викликає callback-функцію, передану під час створення – зазвичай, це метод handleMoving батьківського контейнера. Така передача подій реалізує принцип спостерігача (Observer): повзунок повідомляє свій контейнер про зміну, а контейнер вже реагує, оновлюючи підлеглі елементи. У нашій реалізації метод CollageSeparator.initialize() приймає на вході callback і встановлює його на подію 'moving'. Завдяки цьому під час пересування будь-якого повзунка автоматично запускається відповідний алгоритм перерозподілу розмірів зображення.

Точність та цілісність композицій колажу. Ще одним важливим результатом є те, що всі отримані колажі зберігають високий рівень акуратності: жоден елемент зображення не виходив за межі свого контейнера, не виникало накладень або небажаних проміжків. Алгоритми контролю мінімальних розмірів контейнерів зображення успішно запобігли ситуаціям, коли зображення могли б стиснутися до надто малого розміру або зникнути. Наприклад, коли випробовувалось екстремальне стиснення стовпця з трьома фотографіями, повзунок блокувався саме в той момент, коли висота найменшої фотографії досягала 50 px, і не давав змоги зменшити її далі – це підтвердило коректність реалізації обмежень. Отже, модель колажу гарантує, що якими б не були дії користувача, результат залишатиметься в межах зада-

них допустимих параметрів (мінімальні розміри, загальні розміри площини тощо). Це важливо для професійного використання, де потрібно дотримуватися стандартів макету.

Обмеження шаблонних моделей у сучасних редакторах графічних зображень. Розповсюджені редакто-

ри колажів, такі як Photoshop Online [1] (рис. 9), Canva [3] (рис. 10) та Pixlr Editor [15] (рис. 11), використовують переважно шаблонні моделі компоновання колажу, де елементи зображення розміщуються у фіксованих рамках, що обмежує їх гнучкість та автоматизацію процесу створення колажів.

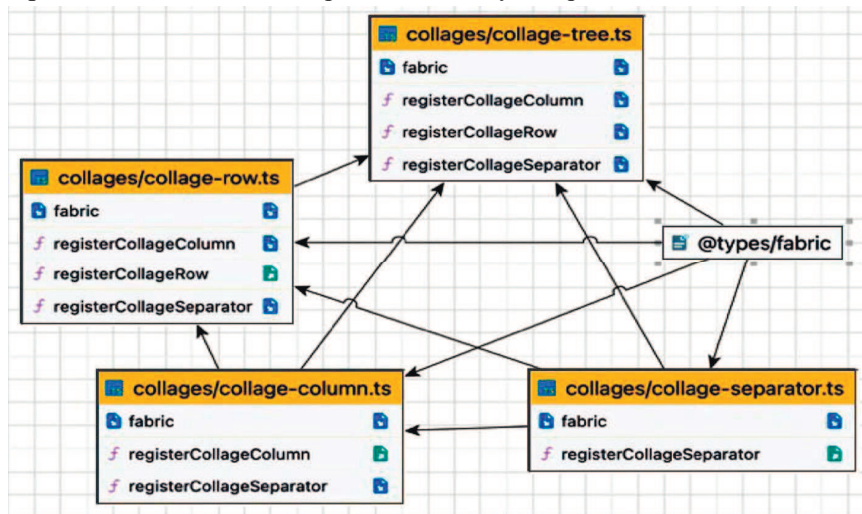


Рис. 8. Діаграма класів редактора колажів / Class diagram of the collage editor

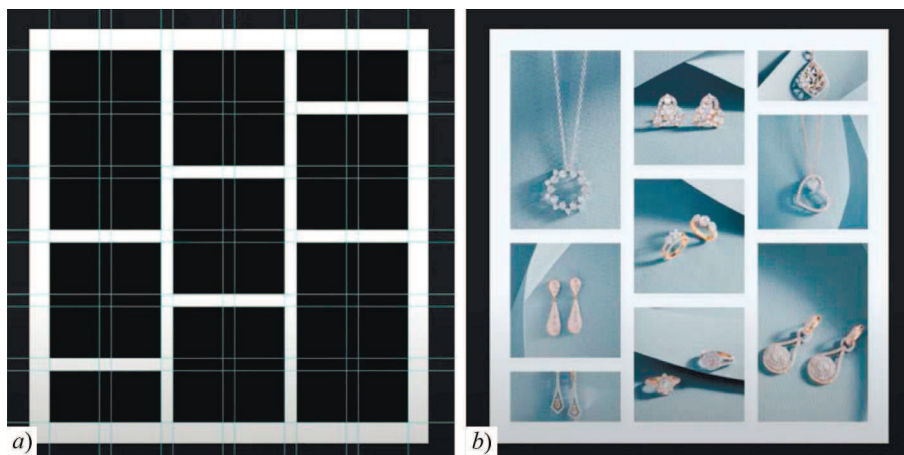


Рис. 9. Полотно із сіткою та контейнерами (a), Колаж зображення (b) / Canvas with grid and containers (a), Image collage (b)

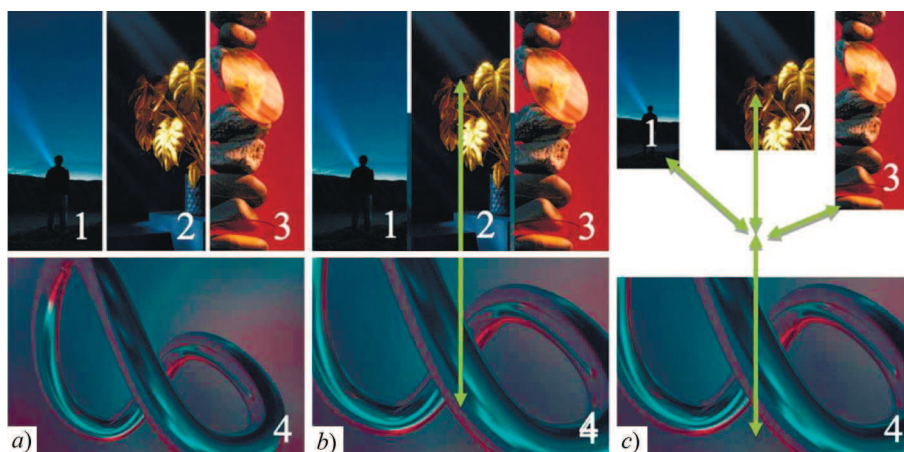


Рис. 10. Колаж у початковому вигляді (a), накладання фрагментів зображень (b), ілюстрація відсутності зв'язків між елементами колажу (c) / Collage in the initial state (a), overlapping of image fragments (b), illustration of the absence of connections between collage elements (c)

Отже, результати дослідження підтверджують ефективність запропонованої моделі колажу: зменшує трудомісткість його редагування, забезпечує узгодженість композиції колажу (автоматичні пропорційні зміни) і є достатньо продуктивною для інтерактивного застосування навіть на вебплатформах. Це дає змогу рекомен-

дувати її для впровадження у реальні редактори графічних зображень.

Обговорення результатів дослідження. У дослідженні [7] автори навели модель VTN (англ. *Variational Transformer Network*), яка поєднує варіаційні автокодекери та трансформери для автоматичного компоновання

графічних елементів макету колажу. Автори дослідили використання механізму самоуваги для аналізу просторових взаємозв'язків між елементами макету колажу, що дає змогу їх генерувати з високою варіативністю та відповідністю до заданих дизайнерських правил без прямого нагляду.

У дослідженні [9] автори розглянули застосування методів штучного інтелекту для автоматичного створення



Рис. 11. Колаж у початковому вигляді (а), ілюстрація зміни розмірів елемента колажу з № 1, та відсутності зв'язку між елементами з № 2, 3, 4 (b) / Collage in the initial state (a), illustration of resizing collage element № 1 and the absence of connection between elements № 2, 3, 4 (b)

У дослідженні [12] розроблено систему Flowy, яка аналізує великі набори інтерфейсів користувача, автоматично визначає дизайн-патерни і формує анотації до них. Автори дослідили ефективність використання штучного інтелекту для виявлення та систематизації дизайн-патернів, що допомагає дизайнерам приймати обґрунтовані рішення та підтримувати послідовність під час розроблення багатосторінкових інтерфейсів.

У дослідженні [17] автори навели модель Visual Layout Composer, яка поєднує векторні та піксельні дифузійні моделі для створення графічних композицій зображень. Досліджено, як інтеграція двох типів даних – векторної графіки і растрових зображень – дає змогу досягти високої точності та деталізації в автоматично згенерованих композиціях зображення, забезпечуючи гнучкість у роботі дизайнерів.

У дослідженні [8] розроблено систему AutoPhoto, яка використовує метод глибокого навчання з підкріпленням для автоматичного компоновання фотографій із заданими естетичними характеристиками. Автори проаналізували ефективність навчання агента на великому наборі даних із заздалегідь визначеними критеріями естетики, продемонструвавши високу якість автоматично створених композицій та їх близькість до професійних стандартів фотографії.

Також, на відміну від системи AutoPhoto [8], де основний акцент зроблено на повністю автоматичну естетичну оцінку та компоновання зображень, модель колажу, яку ми пропонуємо, забезпечує баланс між автоматизацією та ручним керуванням, даючи можливість користувачу самостійно редагувати макет, гарантуючи при цьому автоматичну адаптацію сусідніх елементів зображення для підтримки цілісності структури колажу.

У дослідженні [21] запропоновано модель Composition-Aware Graphic Layout GAN (CGL-GAN), орієнтовану на створення візуально-текстових презентацій. Автори проаналізували вплив просторових і семантичних характеристик зображень на результати генерування

макетів постерів. Модель створення макетів постерів, яку запропонували містить двоетапний підхід, де спочатку визначається базовий шаблон елементів, а потім використовується просторовий трансформер для уточнення композиції зображення відповідно до класичних дизайнерських принципів, таких як золоте січення, що дало можливість значно підвищити візуальну якість макетів постерів.

макетів графічних об'єктів, а також розробили спеціальний модуль вирівнювання доменів для усунення розбіжностей між навчальними та тестовими даними, що дало змогу значно покращити якість отриманих композицій зображень.

Порівняння отриманих результатів із наведеними вище дослідженнями свідчить, що деревоподібна модель колажу з поперечними зв'язками, яку ми розробили, істотно відрізняється за структурною організацією. Зокрема, наша модель колажу базується на чітко визначеній чотирирівневій ієрархії, що забезпечує чіткий контроль над композицією і високу інтерактивність редагування, чого немає у дослідженнях [7, 9] та [21], які переважно орієнтовані на автоматичну генерацію без подальшого ручного редагування.

На відміну від системи Flowy [12], яка сфокусована на аналізі дизайн-патернів та підтримці прийняття рішень на абстрактному рівні, наша модель колажу надає безпосередні інструменти інтерактивного компоновання графічних елементів зображення з автоматичним підтриманням композиційної узгодженості.

Порівняно з Visual Layout Composer [17], наша модель колажу демонструє перевагу завдяки наявності механізмів автоматичного оновлення суміжних елементів, які забезпечують гнучкість і зручність за інтерактивного редагування композицій колажу. Це дає змогу значно скоротити час створення та редагування колажів, підвищуючи ефективність дизайнерської роботи.

У дослідженні [13] запропоновано інструмент для смартфонів, призначеного для створення фотоколажів. Автори представили застосунок PhotoPieces – програмний інструмент, який дає змогу користувачам створювати колажі з фотографій на льоту на портативному пристрої, оснащеному камерою, використовуючи попередньо визначені або власні шаблони. Користувачі також можуть створювати власні шаблони на льоту, роблячи фотографію та перетворюючи її на шаблон зображення з кількома регіонами. Було проведено попереднє дослід-

дження за участю різних типів користувачів, щоб оцінити його цінність як творчого інструменту.

Розповсюджені редактори колажів, такі як Photoshop Online [1] (рис. 13), Canva [3] (рис. 14) та Pixlr Editor [15] (рис. 15), використовують переважно шаблонні моделі компонування, де елементи розміщуються у фіксованих рамках, що обмежує їх гнучкість та автоматизацію процесу створення колажів.

Запропонована деревоподібна модель колажу з поперечними зв'язками його елементів має низку переваг, що відрізняють її від традиційних моделей. Передусім, вона забезпечує високу гнучкість і інтерактивність під час редагування композиції графічного зображення. Користувач може вільно експериментувати зі структурою колажу – додавати або видаляти рядки й стовпці, змінювати їх розміри – не турбуючись про наслідки для суміжних елементів, оскільки система сама підтримує цілісність структури колажу.

Отже, внаслідок виконаної роботи можна сформулювати такі наукову новизну та практичну значущість результатів дослідження.

Наукова новизна отриманих результатів дослідження – удосконалено метод інтерактивного редагування та впорядкування графічних композицій, що полягає у використанні чотирирівневої деревоподібної моделі колажу з динамічними взаємозв'язками між рівнями та вершинами для підтримки цілісності та узгодженості елементів зображення у реальному часі.

Практична значущість результатів дослідження – запропонований метод інтерактивного редагування та впорядкування графічних композицій можна застосовувати у професійній діяльності дизайнерів, фотографів, контент-менеджерів, UI/UX-спеціалістів і освітян, що дасть їм змогу створювати якісно оформлені колажі, перетягуючи зображення на дизайнерські шаблони і миттєво завантажувати їх на будь-який пристрій.

Висновки / Conclusions

Розроблено деревоподібну модель колажів з поперечними зв'язками між їхніми елементами, яка дає змогу інтегрувати автоматичні алгоритми їх компонування з інтерактивними процедурами редагування, а також значно підвищує зручність управління отриманими композиціями завдяки структурній узгодженості та взаємозалежності їхніх елементів. За результатами виконаного дослідження можна зробити такі основні висновки.

1. Встановлено, що відомі моделі оброблення колажів не мають взаємопов'язаних зв'язків між їхніми елементами, що істотно ускладнює редагування колажів.
2. Розроблено чотирирівневу деревоподібну модель колажів з поперечними зв'язками, яка автоматично пов'язує елементи колажу так, що зміна будь-якого його елемента відповідно перебудовує пов'язані з ним сусідні елементи і цим самим зменшує затрати часу на редагування колажу.
3. Зверифіковано і підтверджено коректність створеної деревоподібної моделі колажу з поперечними зв'язками і моделей функціонування системи – зокрема, моделей додавання, видалення, масштабування, автоматичного узгодження та редагування графічних елементів колажу, у спосіб реалізації відповідного програмного забезпечення на основі мови програмування TypeScript, фреймворку Angular та бібліотеки FabricJS на підставі HTML5 Canvas технологій.
4. Встановлено, що впровадження розробленої деревоподібної моделі колажів можна ефективно застосувати

для автоматизації процесів створення графічних колажів у діяльності дизайнерів, фотографів, контент-менеджерів, UI/UX-спеціалістів та в освітній практиці.

References

1. Adobe Photoshop. (n.d.). Photoshop.adobe.com. URL: <https://photoshop.adobe.com/discover>
2. Angular. (2025). Angular.dev. URL: <https://angular.dev/api>
3. Canva. (2025). Canva. URL: <https://www.canva.com/>
4. Canvas API. (2019, December 2). MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API
5. Cao, Y., Ma, Y., Zhou, M., Liu, C., Xie, H., Ge, T., & Jiang, Y. (2022). Geometry Aligned Variational Transformer for Image-conditioned Layout Generation. *Proceedings of the 30th ACM International Conference on Multimedia*, 1, pp. 1561–1571. <https://doi.org/10.1145/3503161.3548332>
6. Chen, L., Jing, Q., Tsang, Y., & Zhou, T. (2024). Iris: a multi-constraint graphic layout generation system. *Frontiers of Information Technology & Electronic Engineering*, 25(7), 968–987. <https://doi.org/10.1631/fitee.2300312>
7. Fernández, D. P., Postels, J., & Tombari, F. (2021). Variational Transformer Networks for Layout Generation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10163–10173. <https://doi.org/10.1109/cvpr46437.2021.01343>
8. Hadi AlZayer, Lin, H., & Bala, K. (2021). AutoPhoto: Aesthetic Photo Capture using Reinforcement Learning. 2021 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 944–951. <https://doi.org/10.1109/iros51168.2021.9636788>
9. Huo, H., & Wang, F. (2022). A Study of Artificial Intelligence-Based Poster Layout Design in Visual Communication. *Scientific Programming*, 2022, 1–9. <https://doi.org/10.1155/2022/1191073>
10. JSDoc: Home. (n.d.). Fabricjs.com. URL: <http://fabricjs.com/docs/>
11. Liu, L., Zhang, H., Jing, G., Guo, Y., Chen, Z., & Wang, W. (2017). Correlation-Preserving Photo Collage. *IEEE Transactions on Visualization and Computer Graphics*, 24(6), 1956–1968. <https://doi.org/10.1109/TVCG.2017.2703853>
12. Lu, Y., Tong, Z., Zhao, Q., Oh, Y., Wang, B., & Li, T. J.-J. (2024). Flowy: Supporting UX Design Decisions Through AI-Driven Pattern Annotation in Multi-Screen User Flows. ArXiv (Cornell University). <https://doi.org/10.48550/arxiv.2406.16177>
13. Marzo, Asier, & Ardaiz, Oscar. (2012). A Smart Phone Tool to Create Photo Collages. *Procedia Computer Science*, Vol. 15, 159–162. <https://doi.org/10.1016/j.procs.2012.10.067>
14. Mozilla. (2019, March 4). WebGL: 2D and 3D graphics for the web. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API
15. Online Photo Editor. Pixlr Editor. (2019). Pixlr.com. URL: <https://pixlr.com/editor/>
16. Rother, C., Bordeaux, L., Hamadi, Y., & Blake, A. (2006). Auto-Collage. *ACM Transactions on Graphics*, 25(3), 847–852. <https://doi.org/10.1145/1141911.1141965>
17. Shabani, M. A., Wang, Z., Liu, D., Zhao, N., Yang, J., & Furukawa, Y. (2024). Visual Layout Composer: Image-Vector Dual Diffusion Model for Design Layout Generation. 2024 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9222–9231. <https://doi.org/10.1109/cvpr52733.2024.00881>
18. Shi, Y., Shang, M., & Qi, Z. (2023). Intelligent layout generation based on deep generative models: A comprehensive survey. *Information Fusion*, 100, 101940–101940. <https://doi.org/10.1016/j.inffus.2023.101940>
19. Yu, J., Chen, L., Zhang, M., & Li, M. (2022). SoftCollage: A Differentiable Probabilistic Tree Generator for Image Collage. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3719–3728. <https://doi.org/10.1109/CVPR52688.2022.00371>
20. Zhang, M., Li, M., Chen, L., & Yu, J. (2021). Aesthetic Photo Collage with Deep Reinforcement Learning. ArXiv (Cornell University). <https://doi.org/10.48550/arXiv.2110.09775>

21. Zhou, M., Xu, C., Ma, Y., Ge, T., Jiang, Y., & Xu, W. (2022). Composition-aware Graphic Layout GAN for Visual-Textual Presentation Designs. *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pp. 4995–5001. <https://doi.org/10.24963/ijcai.2022/692>
22. Zhou, Y., Jing, Q., Li, Z., Shi, L., Chen, L. & Sun, L. (2023). Element-Conditioned GAN for Graphic Layout Generation. *Neuro-computing*, 591, article ID 127730. <https://doi.org/10.ssm.4648075>

R. I. Kordiuk¹, O. V. Ovsyak²

¹ Lviv Polytechnic National University, Lviv, Ukraine

² Ukrainian National Forestry University, Lviv, Ukraine

TREE-BASED COLLAGE MODEL WITH TRANSVERSE-CONNECTIONS

An interactive tree-based collage model with transverse connections is presented, which is developed to effectively overcome the limitations inherent in existing graphic editors such as Photoshop Online, Canva, and Pixlr Editor. Based on a comprehensive analysis of popular existing solutions, it has been established that the main drawbacks of these collage editors include the absence of automatic connections between image element containers and the necessity for extensive manual adjustments of the collage structure when editing respective compositions. Consequently, some issues regarding maintaining visual consistency and harmony arise when resizing images or altering the positions of individual collage elements. The developed collage model is structured as a clearly defined hierarchical tree with four nested levels: root element, rows, columns, and image element containers. A significant enhancement is the introduction of transverse connections between adjacent containers, which enable automatic bidirectional recalculations of container positions and sizes upon changing the parameters of any individual element. Mathematical foundations of this model are described in detail, specifically utilizing graph and tree theory principles, which formally define the logic behind element interactions within the collage. The implementation of the model is performed using the TypeScript programming language and Fabric.js library, actively employing Composite design patterns for structuring components and Observer patterns for efficient management of reactivity and state changes in real-time. Experimental studies of the developed model have demonstrated significant improvement in the graphic composition editing process: the number of necessary user actions is reduced several times compared to traditional models. Additionally, the system exhibits high performance, responsiveness to changes, and stable maintenance of visual harmony even under intensive element editing. It has been established that the proposed collage model successfully combines the advantages of automated intelligent solutions with the flexibility and precision of manual editing, making it an effective tool for a wide range of users.

Keywords: planar collage; tree structure; collage graphic editor; handleMoving algorithm; image composition.