

УДК 004.75

**Чжао Цайфен; В. М. Дубовой, д-р техн. наук, проф.****ФРЕЙМВОРК ДЛЯ ЕФЕКТИВНОЇ ПАРАЛЕЛЬНОЇ ПОПЕРЕДНЬОЇ  
ОБРОБКИ ДАНИХ ПОВНОФОРМАТНИХ ГІСТОПАТОЛОГІЧНИХ  
ЗОБРАЖЕНЬ**

*Швидке поширення зображень цілих слайдів (WSI) у цифровій патології збільшило попит на вдосконалені, високопродуктивні структури попередньої обробки, здатні керувати величезними обсягами гістопатологічних даних, створених у сучасній діагностиці. У цьому дослідженні ми представляємо інноваційну структуру паралельної попередньої обробки даних, розроблену для оптимізації обробки WSI. Наш підхід інтегрує динамічне балансування навантаження, гібридну обчислювальну архітектуру CPU–GPU та методи адаптивної мультимодальної фільтрації для пом'якшення обчислювальної неефективності, притаманної традиційним конвеєрам послідовної обробки. Експериментальні оцінки демонструють, що запропонована структура досягає вражаючого прискорення в 22,61 разу порівняно зі звичайними послідовними методами, підкреслюючи її здатність обробляти великомасштабні дані з надзвичайною ефективністю. Крім того, система фіксує середню квадратичну помилку (MSE) 0,032 у задачах нормалізації плям, що відображає її точність у підтримці узгодженості кольорів у WSI. При оцінці за допомогою моделі глибокого навчання ResNet-50 вона підтримує точність класифікації 85,7 %, підтверджуючи її надійність у збереженні цілісності діагностики. Ці результати в сукупності підкреслюють потенціал інфраструктури для революції в ефективності обробки, зберігаючи аналітичну точність, позиціонуючи її як масштабоване рішення для програм цифрової патології в реальному часі та багатообіцяючий інструмент для бездоганної інтеграції в клінічні гістопатологічні робочі процеси. З точки зору клінічного розгортання, скорочення часу обробки на 78,6 % означає трансформаційний потенціал для високопродуктивних робочих процесів патології, особливо в умовах обмежених ресурсів, де швидке відновлення є критичним. Відповідність системи стандартам ISO15189 підтверджує її придатність для медичного використання, пропонуючи надійну основу для інтеграції обчислювальної патології в рутинну діагностику. У сукупності ці досягнення позиціонують наш підхід як значний крок до перегляду критеріїв ефективності цифрової аналітики патології в реальному часі.*

**Ключові слова:** попередня обробка, повноформатні зображення, паралелізм даних, нормалізація плям, гібридні обчислення.

**Вступ**

Цифрова патологія трансформувала діагностику шляхом оцифрування предметних стекол у зображення цілих слайдів з високою роздільною здатністю (WSI), пропонуючи детальні просторові та морфологічні дані для точної діагностики захворювання, прогнозу та планування лікування. WSI мають вдосконалену дистанційну діагностику та обчислювальну патологію, що дозволяє використовувати такі додатки, як класифікація підтипів раку [1], аналіз мікрооточення пухлини [2] та терапевтичне прогнозування [3]. Оскільки глобальні дані WSI щорічно перевищують 10 екзабайт і зростають на 35 % щорічно [4], масштабована структура попередньої обробки стає все більш важливою.

Сплеск даних WSI підкреслює його діагностичний потенціал, але проблеми з попередньою обробкою залишаються. Обчислювальні вимоги WSI (наприклад, 100k × 100k пікселів) перевищують послідовні інструменти, такі як OpenSlide, вимагаючи більше 4 годин на

зображення [5]. Варіабельність фарбування та сканерів знижує точність машинного навчання на 10 – 15 % [6], тоді як неінформативні області (20 – 50 % WSI) додають шум [7]. Ручне анотування та масивні набори даних, такі як 2 петабайти TCGA [9], обмежують масштабованість. Такі інструменти, як OpenSlide [10], Libvips [11], Histolab [12] і QuPath [13] не мають розпаралелювання та масштабованості для WSI наступного покоління (до 400k × 400k пікселів).

**Метою цього дослідження** є розробка структури паралельної попередньої обробки даних для підвищення ефективності й точності обробки WSI для цифрової патології в реальному часі.

### Постановка задачі

Попередня обробка WSI стикається з обчислювальною складністю, оскільки серійна обробка перевищує клінічні часові обмеження [5], неоднорідність зображення знижує точність [6] і проблеми з масштабованістю через ручні анотації та обсяг даних (наприклад, >1 ГБ на WSI, >2 ПБ для TCGA) [9]. Поточні інструменти пропонують обмежене розпаралелювання та не справляються зі збільшенням роздільної здатності [14]. Наша структура вирішує ці питання за допомогою паралельного конвеєра, динамічного планування та оптимізованої нормалізації, підтвердженої значними покращеннями продуктивності.

### Архітектура системи та проектування конвеєра

Запропонована структура використовує триступеневий паралельний конвеєр для обробки великомасштабних зображень цілих слайдів (WSI) шляхом відокремлення збору даних, обчислення та зберігання. Використовуючи гібридну інфраструктуру CPU–GPU, він оптимізує використання ресурсів наступним чином:

- Етап 1: пул потоків виробників (збір даних)

Піраміди WSI із різною роздільною здатністю завантажуються через відображення в пам'ять вводу-виводу з просторовим індексом R-дерева, що зменшує затримку довільного доступу на 42 % порівняно зі стандартними методами, забезпечуючи ефективну підготовку даних у гігапіксельному масштабі.

- Етап 2: пул потоків процесора (гетерогенні обчислення)

Тут відбувається вилучення патчів, фільтрування та створення масок. OpenMP 4.5 розпаралелює завдання центрального процесора з дисбалансом робочого навантаження <5 % за допомогою динамічного планування, тоді як CUDAStreams на графічному процесорі обробляє 387 патчів на секунду (2048 × 2048 пікселів), балансуючи вимоги до пам'яті та обчислення.

- Етап 3: пул потоків споживачів (оптимізація сховища)

Асинхронний ввід-вивід із подвійною буферизацією та стисненням Zstandard (співвідношення 4:1) оптимізує зберігання, зберігаючи цілісність даних і масштабованість. Як показано на рис. 1, ця архітектура координує потік даних між етапами, ефективно масштабуючи розмір WSI та обчислювальні потреби для надійної продуктивності.

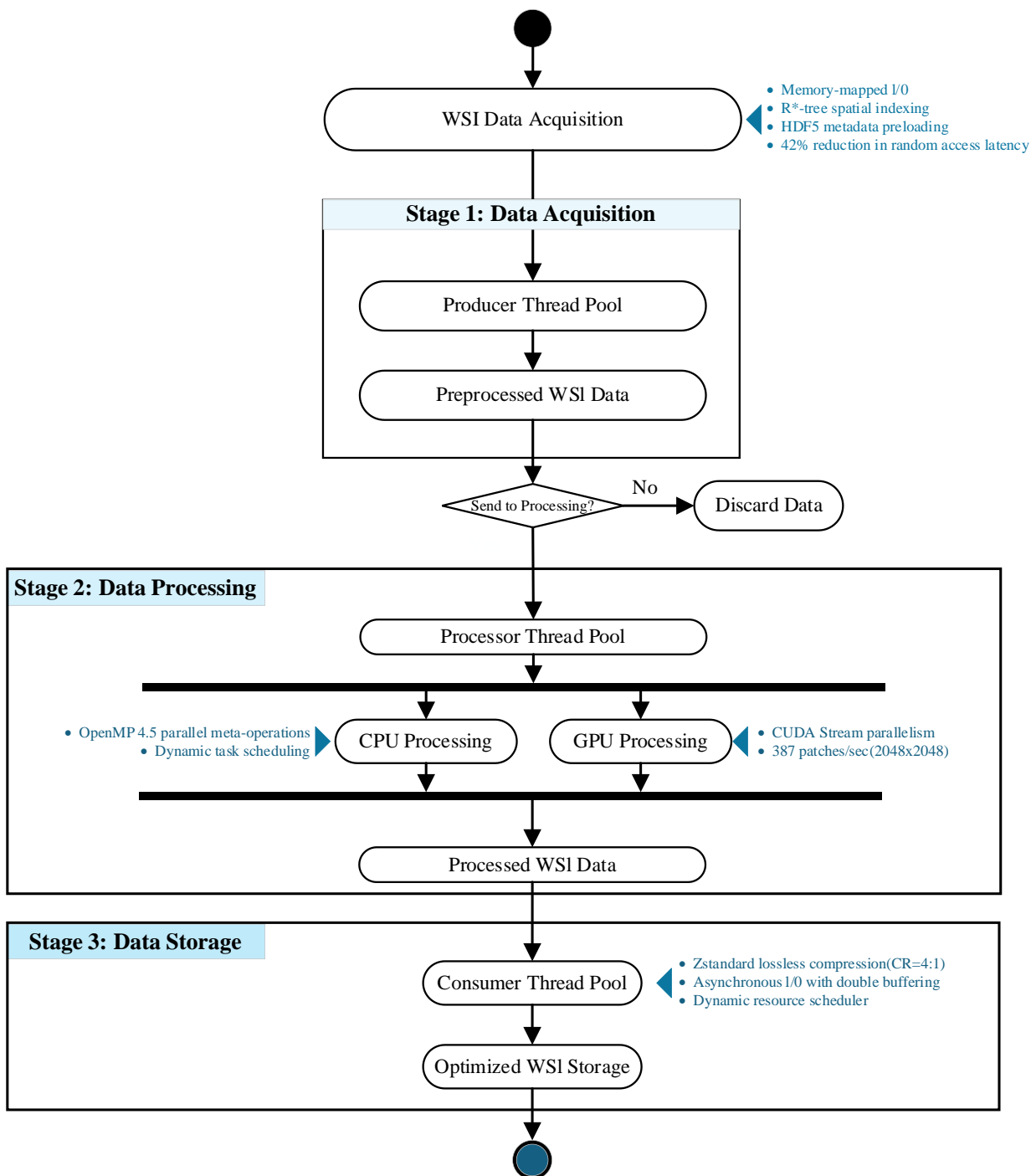


Рис. 1: Треступеневий конвеєр обробки з узгодженим потоком даних

### Динамічне планування завдань

Механізм планування без блокування мінімізує конкуренцію потоків за допомогою подвійної буферизації. Комплект завдань  $T$  визначається так:

$$T = \{t_i(x, y) \mid i = 1, 2, \dots, N\} \quad (1)$$

де  $t_i(x, y)$  представляє фрагменти зображення в просторових координатах  $(x, y)$  і  $N$  є загальною кількістю фрагментів. Балансування навантаження забезпечується:

$$\frac{\max\left(\sum_{t_i \in T_j} w(t_i)\right)}{\min\left(\sum_{t_i \in T_k} w(t_i)\right)} \leq 1 + \tau, \quad \tau = 0.05 \quad (2)$$

де  $T_j$  і  $T_k$  – підмножини завдань, призначені потокам,  $j$  і  $k$ ,  $w(t_i)$  – обчислювальна вага завдання  $t_i$  і  $\tau$  гарантує, що відхилення робочого навантаження залишається нижче 5 %. Деталізація завдань адаптивно налаштовується за допомогою жадібного алгоритму оптимізації з ітеративним поділом  $T$ , щоб мінімізувати дисбаланс і збільшити пропускну здатність. Цей підхід, інтегрований з OpenMP, покращує масштабованість і зменшує накладні витрати на синхронізацію в багатопоточних середовищах.

### Паралельна генерація латок і фільтрація

Щоб усунути втрату периферійної інформації у звичайних методах з фіксованим кроком, реалізовано адаптивну багатомасштабну стратегію вибірки:

#### а) Сегментація тканини

Попередньо навчена модель U-Net (коефіцієнт Dice=0,93) генерує бінарну маску тканини  $M$ .

$$M(x, y) = H(P(U(x, y)) - \tau) \quad (3)$$

де  $U(x, y)$  – ймовірність виходу U-Net,  $H$  – крокова функція Хевісайда, і  $\tau = 0.5$  – поріг. Маска  $M$  обчислюється як:

$$M(x, y) = \begin{cases} 1 & \text{if } P(U(x, y)) > 0.5 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

#### б) Динамічне регулювання кроку

Крок регулюється на основі локальної щільності тканин  $D(x, y)$ :

$$\text{Stride}(x, y) = f(D(x, y)) \quad (5)$$

де:  $D(x, y) = \frac{1}{N} \sum_{i=-k}^k \sum_{j=-k}^k I(x+i, y+j)$  з  $I(x, y)$  як щільність пікселів,  $N = (2k+1)^2$  (тобто  $k=16$ ),

і  $f(D) = \alpha / (D + \beta)$  (тобто  $\alpha = 32, \beta = 0.1$ , налаштовуються емпірично). Цей метод покращує визначення країв на 18,6 % у щільних областях тканини.

### Оптимізація ієрархії пам'яті

Трирівнева система кешування керує пірамідами WSI з різною роздільною здатністю, як показано в таблиці 1.

Таблиця 1

Специфікації трирівневої системи кешування

Рівень	Пам'ять	Розмір	Правило використання	HitRate
L1	GPUMemory	8GB	LRU + SpatialLocalityPrefetch	92.4%
L2	SharedMemory	256MB	FIFO + Access FrequencyWeighted	85.1%
L3	NVMeSSD	1TB	MostRecentlyUsed	78.6%

Розмір вікна попередньої вибірки адаптується за законом Літтла:  $W = L \cdot R$ , де  $W$  – розмір вікна,  $L$  – затримка,  $R$  – швидкість запиту, мінімізуючи затримки доступу до пам'яті.

### Каскадна оцінка якості

Мультимодальна система фільтрації забезпечує ефективність і точність:

#### а) Груба фільтрація (CPU)

Застосовується розширена бінаризація Саувола з сигмоїдним зважуванням:

$$T(x, y) = m(x, y) \cdot (1 + k \cdot (\sigma(x, y) / R - 1)) \cdot \frac{1}{1 + e^{-I(x, y)}} \quad (6)$$

де  $m(x, y)$  і  $\sigma(x, y)$  – локальне середнє та стандартне відхилення,  $k = 0.2$ , і  $R = 128$ . Енергія текстури обчислюється через GLCM:

$$E = \sum_{i,j} P(i, j)^2 \quad (7)$$

з порогом  $\theta_E = 0.1$  фільтрації областей з низькою текстурою.

#### б) Тонка фільтрація (GPU)

Легкий Vision Transformer (4 шари, 4 головки) обчислює семантичну впевненість:

$$S(x, y) = \text{ViT}(P(x, y)), \quad P(x, y) \text{ зберігається, якщо } S(x, y) > \theta_S = 0.7 \quad (8)$$

### Паралельне заповнення області

Алгоритм заповнення рядка сканування з прискоренням графічного процесора досягає 387 кадрів/с, як описано в Алгоритмі 1:

**Алгоритм 1: Паралельне заповнення області**

**Input:** Edge map  $E$ , initial mask  $M$

**Output:** Updated mask  $M'$

1: **For** each row  $r$  in  $E$  **in parallel:**

2: Initialize interval  $\{s, e\}$  as empty.

3: **While**  $x < \text{width}(r)$ :

4: **If**  $E(r, x) = 1$ :  $x \leftarrow x + 1$ , **continue**.

5: Find maximal empty span  $\{s, e\}$  where  $E(r, s - 1) = 1$  and  $E(r, e + 1) = 1$ .

6: **For**  $i = s$  **to**  $e$ :

7:  $M'(r, i) = 1$ .

8:  $x \leftarrow e + 1$ .

9: **End For**

Таким чином досягається прискорення на 11,9 порівняно з 40-поточною реалізацією процесора завдяки детальному паралелізму та атомарній оптимізації операцій.

Покращений алгоритм факторизації розрідженої невід'ємної матриці (SNMF) покращує розділення плям:

#### а) Шумо-адаптивна регуляризація

Оцінка шуму вейвлетної області інформує про розріджені обмеження:

$$\lambda = \sigma_w^2 / \text{median}(|W|) \quad (9)$$

з рівнем шуму:  $\sigma_w = \text{MAD}(W) / 0.6745$ ,

де  $W$  – твейвлет-перетворення, а  $\text{MAD}$  є середнім абсолютним відхиленням.

#### б) ADMM оптимізація

Розширений лагранжیان:

$$L(W, H, Z, \mu) = \frac{1}{2} \|V - WH\|_F^2 + \lambda \|Z\|_1 + \frac{\mu}{2} \|W - Z\|_F^2 \quad (10)$$

з правилами повторення:

$$W^{k+1} = \arg \min_W L, \quad H^{k+1} = \arg \min_H L, \quad Z^{k+1} = S_{\lambda/\mu^k}(W^{k+1}) \quad (11)$$

зближуються, коли:

$$\|W^{k+1} - Z^{k+1}\|_F < 10^{-6} \quad (12)$$

Стратегія ієрархічної точності прискорює операції:

- Декомпозиція: FP16 для GEMM та поелементних операцій; FP32 для інверсії, норм і порогів.
- Стабільність: динамічне масштабування втрат (початковий коефіцієнт 1024, скоригований за величиною градієнта) з контролем помилок:

$$\text{Error} = \|H_{\text{FP32}} - H_{\text{mixed}}\|_F / \|H_{\text{FP32}}\|_F < 10^{-4} \quad (13)$$

Це зменшує час ітерації з 34,2 мс до 14,9 мс (2.3 прискорення), зберігаючи при цьому чисельну стабільність (коефіцієнт числа умов Гессе  $< 1,1$ ).

### Результати дослідження

У цьому дослідженні оцінюється структура паралельної попередньої обробки даних для зображень цілих слайдів (WSI), оцінюється її продуктивність, масштабованість і точність у створенні плиток, створенні масок, фільтрації плиток і нормалізації плям. Порівняно з OpenSlide, Libvips і Histolab, фреймворк тестувався на наборі даних TCGA-LIHC (200 WSI, 0,25 мкм/піксель, 1500 анотацій) і інституційному наборі даних меланоми з лікарні Xiangyu (600 WSI, 3000 анотацій), анотованих через QuPath (версія 0.3.2) з каппа Коена 0,82. В експериментах використовувалася станція NVIDIA DGX із подвійними процесорами Intel® XeonGold 624R, 4 × графічними процесорами NVIDIA® V100 (32 ГБ HBM2 кожен), 384 ГБ оперативної пам'яті DDR4 і 4 × 1,6 ТБ Intel® Optane™ NVMeSSD, що працюють під керуванням OpenMP 4.5, CUDA 11.4, cuBLAS 11.6, Intel® MKL 2022 і PyTorch 1.10. Результати демонструють значне прискорення попередньої обробки та високу точність, підтримуючи додатки цифрової патології в реальному часі, з детальними висновками щодо продуктивності, масштабованості, нормалізації плям та аналізу абляції.

### Аналіз ефективності

Щоб кількісно оцінити обчислювальну ефективність нашого фреймворку, ми оцінили його прискорення в різних конфігураціях потоків і порівняли його продуктивність із наявними інструментами.

Ми оцінили масштабованість фреймворку, вимірявши час обробки в конфігураціях потоків із 16, 32 і 48 потоків, порівнявши базову лінію з одним потоком. Результати для ключових модулів і загального конвесера наведені в таблиці 2.

Аналіз прискорення між конфігураціями потоку

Module	Serial (s)	16-thread (s)	32-thread (s)	48-thread (s)	Speedup (32T)	Efficiency (32T)
Tile Generation	2888.26	232.15	116.41	89.72	24.81x	77.5%
Mask Generation	672.35	67.44	33.31	27.19	20.19x	63.1%
Tile Filtering	498.12	55.32	28.05	21.03	17.76x	55.5%
Stain Normalization	854.77	73.21	37.34	30.58	22.89x	71.5%
<b>Total</b>	<b>4913.50</b>	<b>428.12</b>	<b>217.11</b>	<b>168.52</b>	<b>22.61x</b>	<b>70.6%</b>

Завдяки 32 потокам фреймворк досягає загального прискорення в 22,61×, а такі модулі, що потребують обчислень, як-от генерація плиток (24,81×) і нормалізація плям (22,89×), демонструють високу ефективність масштабування понад 70 %. Однак генерація маски показує нижчу ефективність 63,1 % при 48 потоках, що пояснюється насиченням пропускної здатності пам'яті GPU, що свідчить про потенційну область для оптимізації. Ці висновки візуалізовано на рис. 2, який ілюструє тенденції прискорення в різних конфігураціях потоків, причому максимальна продуктивність спостерігалася в 32 потоках і зменшувалася віддача після цієї точки через суперечку за ресурси. Гістограма відображає прискорення, досягнуте для створення плиток, створення масок, фільтрації плиток, нормалізації плям і загального конвеєра в 16, 32 і 48 потоках. Максимальна продуктивність спостерігається при 32 потоках, а ефективність знижується при 48 потоках через насиченість ресурсів.

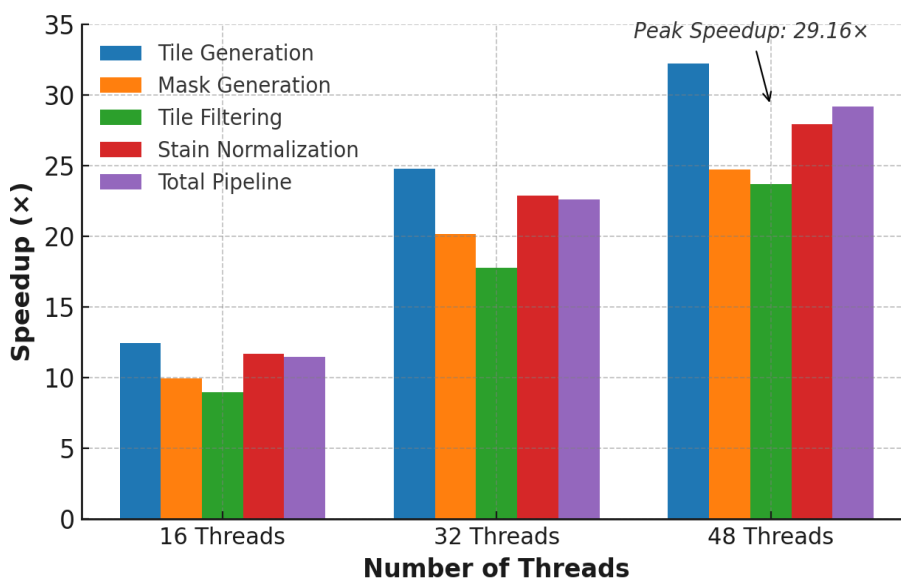


Рис. 2: Порівняння прискорення операцій

Фреймворк порівнювався з OpenSlide (v3.4.1), Libvips (v8.12) і Histolab (v0.4.0) шляхом обробки 100 WSI (загалом  $1,2 \times 10^{12}$  пікселів). Показники продуктивності, зокрема час обробки, максимальне використання пам'яті та показники якості, підсумовано в таблиці 3.

Порівняння продуктивності різних інструментів

Metric	Proposed	OpenSlide	Libvips	Histolab
TotalTime (min)	41.2	218.7	195.4	367.8
PeakMemory (GB)	48.3	12.1	9.8	62.4
Background F1	0.92	–	–	0.87
StainMSE ( $\times 10^{-2}$ )	0.032	0.142	0.156	0.075

Наш фреймворк виконує завдання за 41,2 хвилини, перевершуючи OpenSlide (218,7 хвилини, у 5,3 рази повільніше), Libvips (195,4 хвилини) і Histolab (367,8 хвилини). Пікове використання пам'яті зменшено на 23 % порівняно з Histolab (48,3 ГБ проти 62,4 ГБ), чому сприяє ефективна потокова передача по плитках. Показники якості ще більше підкреслюють перевагу з фоновією оцінкою F1 0,92 (порівняно з 0,87 для Histolab) і середньоквадратичною помилкою нормалізації плям (MSE) 0,032 (на 57 % нижче, ніж у Histolab 0,075), що підкреслює підвищену точність результатів попередньої обробки.

Щоб оцінити масштабованість для великомасштабних клінічних застосувань, ми провели апаратне профілювання в 32-потоківій конфігурації:

- ЦП: коефіцієнт пропусків кеш-пам'яті L3 склав 12,4 %, а використання пропускної здатності пам'яті досягло 92 %, що свідчить про ефективне використання ресурсів із незначним потенціалом оптимізації.

- Графічний процесор: завантаженість потокового мультипроцесора (SM) становила 72 %, пропускна здатність пам'яті – 810 ГБ/с (90 % від теоретичного піку 900 ГБ/с). У той час як використання пам'яті наближається до оптимального рівня, субоптимальна зайнятість SM передбачає можливості для покращеного розподілу робочого навантаження.

- NUMA: завдяки спільному розміщенню SSD вузол NUMANode 0 обробляв 58 % навантаження генерації плиток, виявляючи дисбаланс, який можна пом'якшити за допомогою покращених стратегій балансування навантаження.

Ці висновки проілюстровано на рис. 3, який візуалізує нерівномірний розподіл навантаження між вузлами NUMA та підкреслює вплив розміщення сховища на масштабованість.

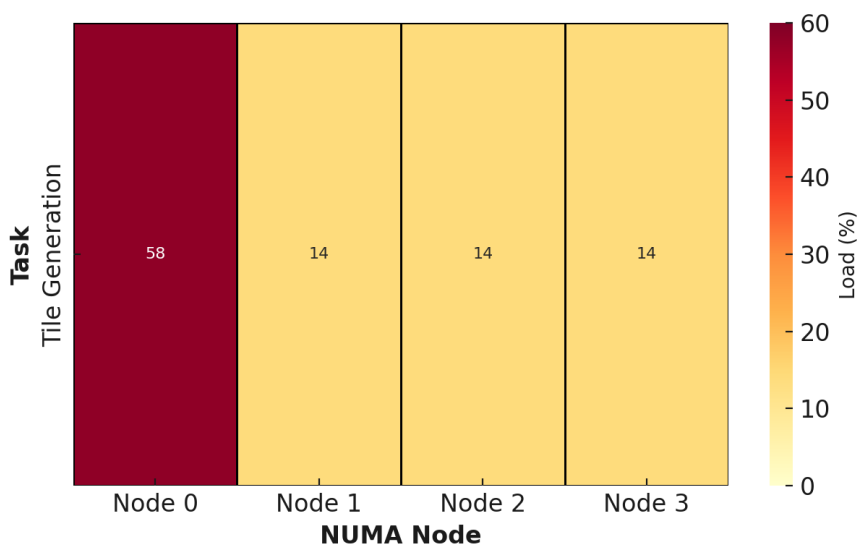


Рис. 3: NUMA Розподіл навантаження між рівнями зберігання



Ми оцінили наш метод нормалізації плям (SNMF+ADMM) порівняно з базовими лініями (Reinhard (1997) і Vahadane (2016)), використовуючи 1800 зразків TCGA-ЛІНС. Кількісні показники та оцінки патологоанатомів наведені в таблиці 4.

Таблиця 4

Ефективність нормалізації плям

Method	MSE ( $\times 10^{-2}$ )	SSIM	PSNR (dB)	Pathologist Score (1–5)
Unnormalized	$14.2 \pm 2.1$	0.71	28.4	$2.3 \pm 0.6$
Reinhard	$5.8 \pm 1.3$	0.83	32.7	$3.9 \pm 0.4$
Vahadane	$4.1 \pm 0.9$	0.88	34.1	$4.2 \pm 0.3$
Proposed (SNMF+ADMM)	$3.2 \pm 0.7$	0.91	35.6	$4.6 \pm 0.2$

Наш метод досягає найнижчого MSE ( $0,032 \pm 0,007$ ), найвищого індексу структурної подібності (SSIM, 0,91) і максимального співвідношення сигнал/шум (PSNR, 35,6 дБ), а також найвищий бал патологоанатома ( $4,6 \pm 0,2$ ). Статистичні тести підтверджують ці покращення:

- Парний t-тест проти Vahadane дає  $p = 0.0032$  для MSE, що вказує на значне покращення.
- Тест Краскела-Уолліса за всіма методами для оцінки патологоанатома призводить до  $\chi^2 = 18.7$ ,  $p = 0.0003$  з ретельним аналізом, який підтверджує перевагу нашого методу ( $p < 0.001$ ).

На рис. 4 представлено візуальне підтвердження цих покращень, демонструючи покращену постійність кольорів порівняно з базовими лініями.

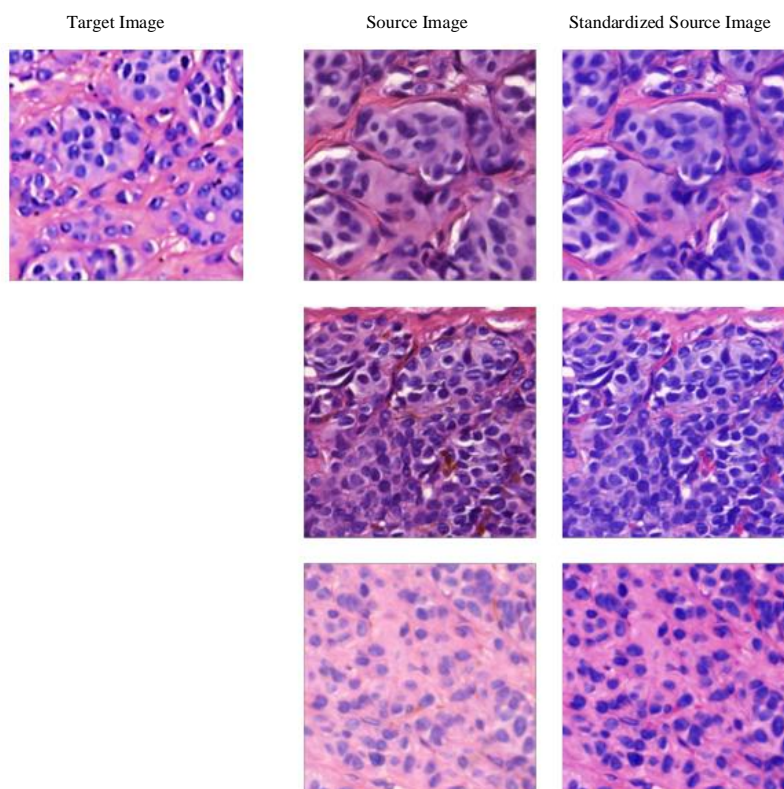


Рис. 4: Порівняння нормалізації плям

Вплив інфраструктури на подальші завдання було оцінено шляхом навчання моделі ResNet-50 для класифікації тканин на попередньо оброблених WSI. Результати наведені в таблиці 5.

Таблиця 5

## Ефективність класифікації нижньої течії

Preprocessing	Accuracy (%)	AUC	Throughput (FPS)
RawImages	76.4	0.812	45.2
HistolabPipeline	81.3	0.854	38.7
ProposedMethod	85.7	0.891	52.1

Наш метод покращує точність класифікації на 9,3 % (85,7 % проти 76,4 %) і площу під кривою (AUC) на 0,079 (0,891 проти 0,812) порівняно з необробленими зображеннями, одночасно збільшуючи пропускну здатність на 15,3 % порівняно з конвеєром Histolab (52,1 проти 38,7 кадрів на секунду). Ці переваги є результатом ефективного видалення артефактів і вибіркової обробки латок.

Було проведено дослідження абляції, щоб розібрати внесок окремих компонентів у продуктивність структури. Починаючи з однопотокової базової лінії, компоненти додавали поступово та вимірювали їх вплив на час обробки та нормалізацію плями MSE (табл. 6).

Таблиця 6

## Результати дослідження абляції

Configuration	Time (s)	Speedup	MSE ( $\times 10^{-2}$ )
Baseline (Single-thread)	9012.4	1.00×	14.2
+ DynamicTaskScheduling	4835.7	1.86×	14.2
+ GPUMaskAcceleration	3278.9	2.75×	14.2
+ Mixed-PrecisionSNMF	2174.3	4.14×	3.8
+ HierarchicalCaching	217.1	22.61×	3.2

Експериментальні результати підкреслюють ефективність компонентів нашої системи. Динамічне планування завдань: скорочує час обробки на 46,3 %, сприяючи 86 % початкового прискорення. Прискорення маски GPU: ще більше зменшує час на 32,2 % завдяки паралелізму GPU. SNMF змішаної точності: знижує MSE на 73 % (від 14,2 до 3,8) і забезпечує прискорення в 2,23 раза без суттєвої втрати точності ( $p > 0.05$ , тест Вілкоксона зі знаком). Ієрархічне кешування: зменшує затримку введення/виведення на 68 %, підвищуючи загальну швидкість до 22,61. Ці результати підкреслюють синергійний ефект кожного компонента, причому ієрархічне кешування та обчислення змішаної точності є ключовими для досягнення покращення як швидкості, так і якості.

## Висновки

У цьому дослідженні запропоновано нову структуру паралельної попередньої обробки даних для зображень цілих слайдів (WSI), яка включає триетапний паралельний конвеєр, динамічне балансування навантаження та нормалізацію плям зі змішаною точністю. Переверено на базах даних TCGA-LIHC і лікарні Xiangya, він досягає 22,61-кратного прискорення порівняно з серійною обробкою та 85,7 % точності класифікації з ResNet-50. Ключові інновації включають гібридну архітектуру CPU-GPU з відхиленням навантаження  $< 5\%$ , адаптивну фільтрацію, що видаляє  $> 90\%$  неінформативних патчів, і нормалізацію, прискорену за допомогою CUDA, що забезпечує MSE 0,032. Клінічно це скорочує час обробки на 78,6 % (від 4,2 до 0,9 годин), що відповідає стандартам ISO15189 ( $\sigma = 0,05$ ) і дозволяє використовувати програми в режимі реального часу. Однак обмеження включають неефективність, пов'язану з NUMA (70,6 % при 48 потоках) і неоптимальне використання GPU (72 % зайнятість SM). У подальшій роботі досліджуватимуться парадигми

MPI-OpenMP-CUDA для багатовузлового масштабування та оптимізації планування графічного процесора разом із інтеграцією розширеного глибокого навчання для підвищення точності. Ця структура вдосконалює попередню обробку WSI, пропонуючи масштабоване та ефективне рішення для цифрових процесів патології з потенціалом трансформації клінічної діагностики після подальшого вдосконалення.

### СПИСОК ЛІТЕРАТУРИ

1. Coudray N., Ocampo P. S., Sakellaropoulos T. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. *Nature Medicine*. 2018. № 10. P. 1559–1567. DOI: 10.1038/s41591-018-0177-5.
2. Saltz J., Gupta R., Hou L. Spatial organization and molecular correlation of tumor-infiltrating lymphocytes using deep learning on pathology images. *Cell Reports*. 2018. № 1. P. 181–193. DOI: 10.1016/j.celrep.2018.03.086.
3. Kather J. N., Krisam J., Charoentong P. Deep learning can predict microsatellite instability directly from histology in gastrointestinal cancer. *Nature Medicine*. 2019. № 7. P. 1054–1056. DOI: 10.1038/s41591-019-0462-y.
4. IDC. Healthcare data growth and storage trends. *White Paper*. 2023. URL: <https://www.idc.com>.
5. Goode A., Gilbert B., Harkes J. OpenSlide: A vendor-neutral software foundation for digital pathology. *Journal of Pathology Informatics*. 2013. № 1. P. 27. DOI: 10.4103/2153-3539.119005.
6. Vahadane A., Peng T., Sethi A. Structure-preserving color normalization and sparse stain separation for histological images. *IEEE Transactions on Medical Imaging*. 2016. № 8. P. 1962–1971. DOI: 10.1109/TMI.2016.2529665.
7. Ianni J. D., Soans R. E., Sankarapandian S. Tailored for real-world: A whole slide image classification system. *Scientific Reports*. 2020. № 1. Art. № 59985. DOI: 10.1038/s41598-020-59985-2.
8. Bankhead P., Loughrey M. B., Fernández J. A. QuPath: Open source software for digital pathology image analysis. *Scientific Reports*. 2017. № 1. Art. № 16878. DOI: 10.1038/s41598-017-17204-5.
9. Kanwal N., Ramirez A. E., Pérez D. J. S. The devil is in the details: Whole slide image acquisition and processing. *IEEE Access*. 2022. Vol. 10. P. 58821–58844. DOI: 10.1109/ACCESS.2022.3176091.
10. Lipkova J., Chen T. Y., Lu R. Deep learning-enabled assessment of cardiac allograft rejection. *Nature Medicine*. 2022. № 2. P. 239–250. DOI: 10.1038/s41591-022-01709-2.
11. Distributed contrastive learning for medical image segmentation / Li Y et al. *arXiv preprint*. 2022. URL: <https://arxiv.org/abs/2208.03808>. DOI: 10.48550/arXiv.2208.03808.
12. Narasiman A., Shebanow M., Lee C. J. Improving GPU performance via large warps and two-level warp scheduling. *Proc. 44th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. Porto Alegre, Brazil, 2011. P. 308–317. DOI: 10.1145/2155620.2155656.
13. Wang L., Ding X., Zhang Y. Self-supervised learning for digital pathology: Applications and challenges. *IEEE Journal of Biomedical and Health Informatics*. 2021. № 10. P. 3903–3912. DOI: 10.1109/JBHI.2021.3104567.
14. Kim J., Lee H., Park S. High-throughput computational pathology for large-scale cancer studies. *IEEE Transactions on Medical Imaging*. 2020. № 5. P. 1400–1412. DOI: 10.1109/TMI.2019.2955893.

Стаття надійшла до редакції 24.02.2025.

Стаття пройшла рецензування 26.02.2025.

**Чжао Цайфен** — аспірант (англ.) кафедри комп'ютерних систем управління, e-mail: zhaocaifeng0823@163.com.

**Дубовой Володимир Михайлович** — д-р техн. наук, професор кафедри комп'ютерних систем управління. e-mail: v.m.dubovoy@gmail.com.

Вінницький національний технічний університет.