IRSTI 28.23.15 DOI: https://doi.org/10.26577/JMMCS1302202613

A. Ospan*, M. Mansurova, A. Sailau, T. Sarsembayeva
Farabi University, Almaty, Kazakhstan
*e-mail: assel.ospan@kaznu.edu.kz

X-CUT++: A LAYOUT-AWARE OCR PIPELINE FOR KAZAKH-LANGUAGE NEWSPAPERS

We address the problem of structural decomposition of complex multi-column Kazakh-language newspaper pages prior to optical character recognition. We propose a hybrid, fully interpretable layout-aware pipeline named X-Cut++, which combines adaptive binarization, smoothed horizontal/vertical projection profiles, morphological dilation, colour-aware region detection in HSV space, a probabilistic Hough fallback for separator lines, and a rule-based post-OCR structural parser that reconstructs the canonical *title/abstract/author/body* article structure. The method is formulated as a cascade of one-dimensional projection cuts with recursive vertical and horizontal subdivision constrained by geometric and area-based thresholds, ensuring deterministic and reproducible segmentation. Experiments on a multi-issue dataset of the newspaper *Egemen Qazaqstan* (5 issues, Jan–Feb 2024, 72 editorial pages, 300 DPI) demonstrate that X-Cut++ consistently decomposes full pages into coherent article-level fragments. The system produces 230 fragments in total (3.19 per page on average). On a manually verified subset of 15 fragments, the structural parser achieves perfect extraction of titles and abstracts, and correctly identifies all present author lines, confirming the reliability of the post-OCR structural reconstruction.
Key words: layout analysis, projection profiles, adaptive thresholding, HSV color segmentation, Hough transform, OCR, low-resource languages, Kazakh, Tesseract, document structure recovery.

Ә. Оспан*, А. Сайлау, М. Мансурова, Т. Сарсембаева
Әл-Фараби атындағы Қазақ ұлттық университеті, Алматы, Қазақстан
*e-mail: assel.ospan@kaznu.edu.kz

X-Cut++: қазақ тіліндегі газеттерді тануға арналған құрылымға сезімтал OCR конвейері

Оптикалық таңбаларды тану кезеңіне дейін күрделі көпбағаналы газет макеттерін құрылымдық декомпозициялау мәселесі қарастырылады. X-Cut++ деп аталатын гибридті, толық интерпретацияланатын конвейер ұсынылады. Ол адаптивті бинаризацияны, тегістелген горизонталь және вертикаль проекциялық профильдерді, морфологиялық кеңейтуді, HSV кеңістігіндегі түстік аймақтарды анықтауды, бөлгіш сызықтарды табуға арналған Хафтың ықтималдық түрлендіруін және мақаланың канондық құрылымын (*тақырып / аннотация / автор / негізгі мәтін*) қалпына келтіретін ережеге негізделген пост-OCR модулін біріктіреді. Әдіс геометриялық және аудандық шектеулермен реттелетін бірөлшемді проекциялық кесулер каскады ретінде формалданған, бұл нәтижелердің детерминирленуі мен қайталанғыштығын қамтамасыз етеді. *Egemen Qazaqstan* газетінің көпвыпусты деректер жиынында (5 шығарылым, 2024 ж. қаңтар–ақпан, 72 редакциялық бет, 300 DPI) жүргізілген тәжірибелер X-Cut++ әдісінің толық беттерді тұрақты түрде мақала деңгейіндегі құрылымдық фрагменттерге бөлетінін көрсетті. Жүйе барлығы 230 фрагмент (бір бетке орташа 3.19) қалыптастырды. Қолмен тексерілген 15 фрагментте құрылымдық модуль тақырыптар мен аннотацияларды толық дұрыс анықтап, барлық бар автор жолдарын дәл қалпына келтірді, бұл пост-OCR құрылымдық реконструкцияның сенімділігін растайды.
Түйін сөздер: макет талдауы, проекциялық профильдер, адаптивті бинаризация, HSV сегментация, Хаф түрлендіруі, OCR, ресурсы шектеулі тілдер, қазақ тілі, Tesseract, құжат құрылымын қалпына келтіру.

Ә. Оспан*, А. Сайлау, М. Мансурова, Т. Сарсембаева
Казахский национальный университет имени аль-Фараби, Алматы, Казахстан
*e-mail: assel.ospan@kaznu.edu.kz

X-Cut++: каскадно-ориентированный OCR-конвейер для распознавания казахоязычных газет

Рассматривается задача структурной декомпозиции сложных многостраничных газетных макетов казахскоязычных изданий перед этапом оптического распознавания текста. Предлагается гибридный, полностью интерпретируемый конвейер **X-Cut++**, объединяющий адаптивную бинаризацию, сглаженные горизонтальные и вертикальные проекционные профили, морфологическое расширение, цвето-зависимое выделение областей в пространстве HSV, вероятностное преобразование Хафа для линий-разделителей, а также правило-ориентированный пост-OCR модуль, восстанавливающий каноническую структуру статьи *заголовок / аннотация / автор / основной текст*. Метод формализован как каскад одномерных проекционных разрезов с рекурсивным вертикальным и горизонтальным делением, ограниченным геометрическими и площадными критериями, что обеспечивает детерминированность и воспроизводимость результата. Эксперименты на многовыпусковом наборе данных газеты *Egemen Qazaqstan* (5 выпусков, январь–февраль 2024 г., 72 редакционные страницы, 300 DPI) показывают, что X-Cut++ стабильно декомпозирует полные страницы в структурированные фрагменты уровня статей. Система формирует 230 фрагментов (в среднем 3.19 на страницу). На вручную проверенной выборке из 15 фрагментов структурный модуль обеспечивает полное извлечение заголовков и аннотаций и корректно определяет все присутствующие строки авторов, подтверждая надёжность пост-OCR реконструкции структуры.
Ключевые слова: анализ макета, проекционные профили, адаптивная бинаризация, HSV-сегментация, преобразование Хафа, OCR, малоресурсные языки, казахский язык, Tesseract, восстановление структуры документа.

1 Introduction

Newspaper pages are among the most structurally complex categories of printed documents for automatic analysis. A single A2–A1 page typically contains heterogeneous elements, including multi-column text with variable widths, hierarchical headlines, author lines, photographs, coloured panels, infographics, advertisements, and cross-page continuations. This compositional diversity violates assumptions of standard OCR pipelines: direct page-level processing often produces interleaved outputs across columns and articles, resulting in semantically incoherent text that is difficult to index or process downstream [1, 2].

In recent years, document understanding has been dominated by end-to-end neural architectures such as LayoutLMv3 and Donut, which jointly model visual and textual signals [3, 4]. While these models achieve strong results on benchmarks such as PubLayNet and DocLayNet, their performance depends on large-scale annotated datasets and significant computational resources. For low-resource languages such as Kazakh, these assumptions do not hold. Archival newspaper collections are mainly available as scanned images, layout annotations are scarce, and domain-specific models are virtually absent. In addition, newspaper layouts differ substantially from scientific or business documents, leading to severe domain shift.

This creates a fundamental gap: neural methods are accurate but data-hungry, whereas classical approaches are interpretable and data-efficient but often brittle in heterogeneous

layouts. In particular, projection-based methods (e.g., X–Y cut) fail in the presence of coloured panels, irregular spacing, and embedded graphical elements, which violate their structural assumptions.

To address this gap, we propose **X-Cut++**, a hybrid, fully interpretable pipeline for layout-aware decomposition of newspaper pages designed for low-resource settings. Unlike neural approaches, the method requires no labelled training data, is fully deterministic, and provides pixel-level reproducibility. At the same time, it extends classical projection-based methods via a multi-branch fallback architecture that integrates HSV-based colour segmentation and probabilistic Hough line detection.

Formally, the pipeline is defined as a composition of operators:

$$\Pi_{\text{X-Cut++}} = \Pi_{\text{struct}} \circ \Pi_{\text{ocr}} \circ (\Pi_{\text{split}} \cup \Pi_{\text{rescue}}) \circ \Pi_{\text{block}} \circ \Pi_{\text{pre}}$$

where Π_{pre} denotes adaptive preprocessing, Π_{block} performs morphological block aggregation, Π_{split} implements recursive projection-based decomposition, Π_{rescue} includes HSV colour segmentation and Hough-based line detection, Π_{ocr} performs text recognition, and Π_{struct} reconstructs article-level structure.

The main contributions of this work are as follows: (1) a deterministic layout-aware segmentation pipeline that requires no training data; (2) a hybrid rescue architecture combining projection cuts with HSV and Hough-based corrections; (3) a rule-based post-OCR parser reconstructing canonical article components (title, abstract, author, body); (4) empirical validation on multi-issue *Egemen Qazaqstan* datasets showing stable article-level decomposition across diverse layouts.

Beyond segmentation, the extracted structured articles provide high-quality inputs for downstream document intelligence tasks, including summarisation, information extraction, and retrieval-augmented generation for Kazakh-language applications.

2 Related work

Classical document layout analysis methods [5, 6] are traditionally divided into top-down (recursive page decomposition) and bottom-up (aggregation of elementary components). The X-Y cut method [7] formalises the top-down strategy by building a tree of hierarchical projection profiles; this very approach forms the mathematical core of the present work. Bottom-up algorithms, such as RLSA [8] or Voronoi-diagram based partitioning [9], are efficient on homogeneous text blocks, but their applicability is limited in the presence of non-textual elements like infographics and coloured backgrounds.

Modern neural architectures (PubLayNet [10], DocLayNet [11], LayoutLMv3 [3]) achieve state-of-the-art results in structural analysis of scientific and business documents. However, their adaptation to the newspaper layout for low-resource languages requires large annotated datasets. Mukhamediev et al. [12] investigated YOLO-family detectors for Kazakh press, but reported significant difficulties with coloured panels and spanning articles. Similar problems are known for foreign archival corpora, e.g., ENP [13]. Despite the existence of specialised recognition systems for Kazakh archives [14], publicly available, interpretable layout models remain an open issue.

For the OCR stage, the dominant solution for Kazakh is Tesseract 5 [1, 15] with the `kaz` language model. A promising alternative is PaddleOCR [16]; however, given the constraints of

the experimental environment and the goal of maximum reproducibility, Tesseract was used as the primary OCR engine.

Additional recent works by the same research group address complementary aspects of Kazakh-language document and speech processing. A table extraction pipeline named QURMA was developed for knowledge base population [17], demonstrating the feasibility of structural extraction from Kazakh documents. Fine-tuning of the Wav2Vec2 model for Kazakh speech on a limited corpus has been investigated, showing effective transfer learning [18]. Moreover, the perspective of using LLM agents for enhanced tabular data interpretation has been outlined [19], which directly motivates the need for high-quality structured textual output from OCR pipelines, such as the one presented here.

3 Methodology

3.1 Overall pipeline and segmentation algorithm

The X-Cut++ pipeline processes a high-resolution newspaper scan through a sequence of fully deterministic stages. The core logic is summarised in Algorithm 1, which defines the complete cascade from preprocessing to structural recovery.

Algorithm 1 X-Cut++ cascade segmentation of a newspaper page

Require: Image I, configuration θ
Ensure: Set of final fragments $\mathcal{F}$
$B \leftarrow \text{AdaptiveThreshold}(I, b{=}25, C{=}10)$
$M \leftarrow B \oplus K_{15\times3}$
Compute vertical and horizontal projections
Extract cut centres $\{c_i^v\}, \{c_j^h\}$
Build candidate grid $\mathcal{B}$
for $\beta \in \mathcal{B}$ **do**
    Detect vertical cut
    if rescue needed **then**
        apply Hough or HSV fallback
    end if
    split vertically
    for each vertical sub-block **do**
        detect horizontal cut
        split horizontally
        if valid region **then**
            add to $\mathcal{F}$
        end if
    end for
end for
return $\mathcal{F}$

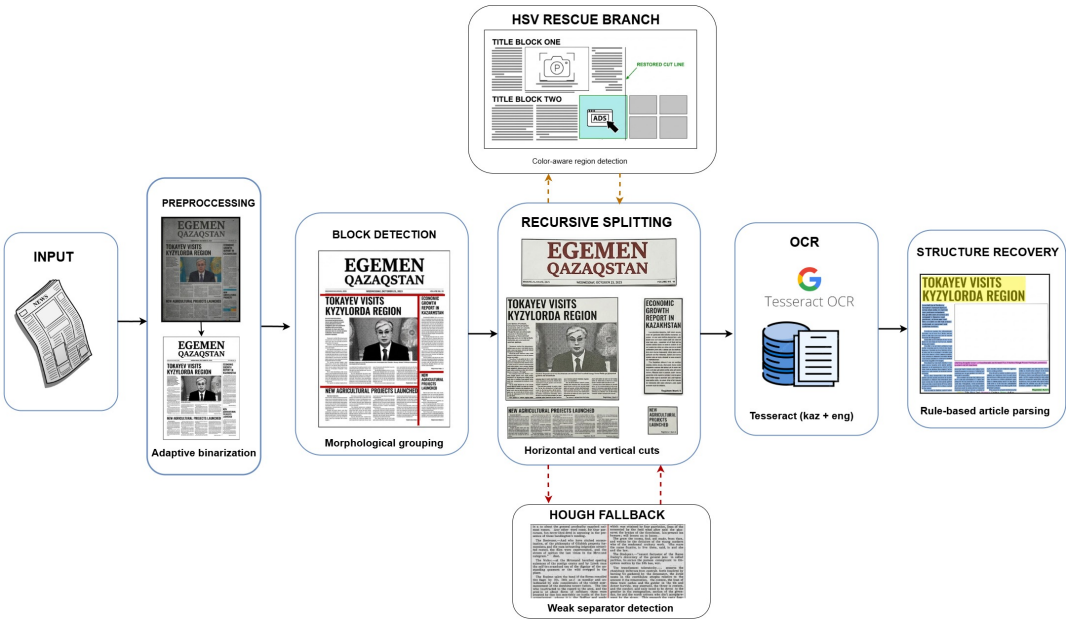

Figure 1: Overall architecture of the X-Cut++ pipeline, showing preprocessing, block detection, recursive splitting with rescue branches, OCR, and structural recovery.

The overall architecture of X-Cut++ is illustrated in Fig. 1.

3.2 Mathematical model of a page

Denote the grey-scale channel of an input page as $I : \Omega \to [0, 255]$, $\Omega = \{1, \dots, W\} \times \{1, \dots, H\}$. For the issues of *Egemen Qazaqstan* scanned at 300 DPI we have $W = 4134$, $H = 6851$, corresponding to a physical page size of 350.01×580.05 mm according to

$$\ell_{\text{mm}} = \ell_{\text{px}} \cdot \frac{25.4}{\text{DPI}}. \tag{1}$$

The first operator of X-Cut++ is pixel-wise adaptive thresholding. Let $W_b(x, y)$ be a square window of radius b centred at (x, y), $|W_b| = (2b + 1)^2$. The local mean

$$T_b(x, y) = \frac{1}{|W_b|} \sum_{(u,v) \in W_b(x,y)} I(u, v) \tag{2}$$

defines the adaptive threshold. The binarised field $B : \Omega \to \{0, 1\}$ is obtained as

$$B(x, y) = \mathbb{1}\big[I(x, y) < T_b(x, y) - C\big], \tag{3}$$

with $b = 25$ (`blockSize`) and $C = 10$. The method `ADAPTIVE_THRESH_MEAN_C` is chosen because newspaper paper is unevenly illuminated during scanning, and a single global Otsu threshold [20] gives poor results on darkened edges.

Next, morphological dilation is applied using a rectangular structuring element $K \in \{0, 1\}^{15 \times 3}$:

$$M = B \oplus K, \qquad (B \oplus K)(p) = \bigvee_{q \in K} B(p + q). \tag{4}$$

The anisotropic shape of the kernel (with a larger horizontal than vertical extent) allows neighbouring characters within the same text line to be merged, while avoiding undesired connections between adjacent lines. This property is crucial for the subsequent projection-based analysis. The effect of the first two processing stages is illustrated in Fig. 2.

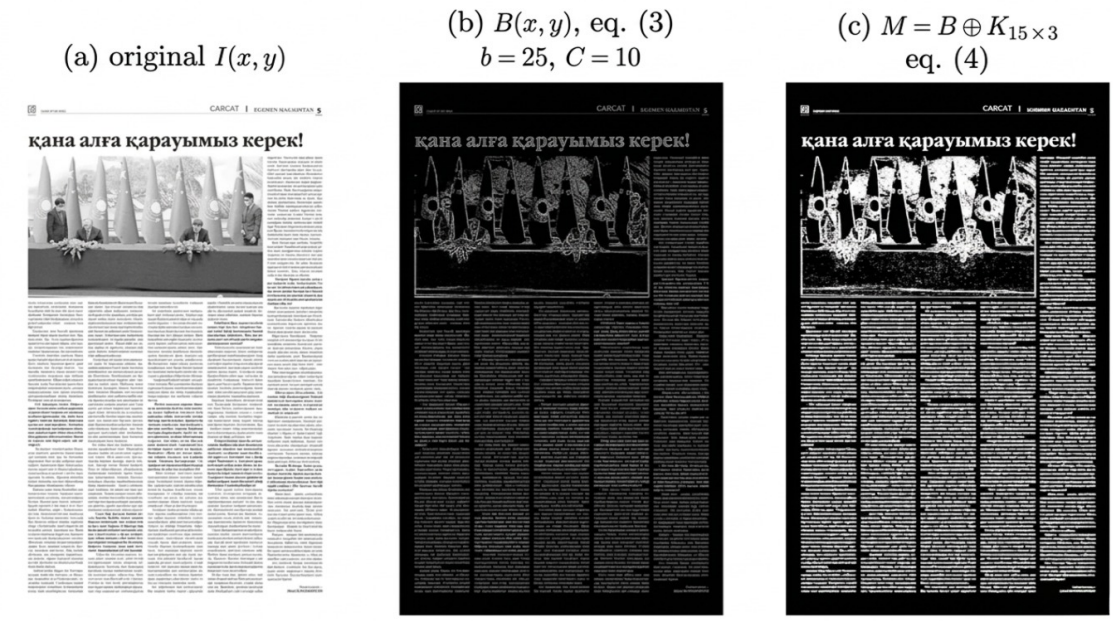

Figure 2: Processing stages of X-Cut++ on a newspaper page: (a) Original grayscale image $I(x, y)$. (b) Binarised field $B(x, y)$ obtained using (3) with parameters $b = 25$ and $C = 10$, where uneven background illumination is suppressed. (c) Result of morphological dilation $M = B \oplus K_{15\times3}$ defined in (4).

Another key component is the use of projection profiles. For the binary field M, the vertical and horizontal projections are defined as

$$P_v(x) = \sum_{y=1}^{H} M(y, x), \qquad P_h(y) = \sum_{x=1}^{W} M(y, x). \tag{5}$$

These projections aggregate foreground pixels along orthogonal axes, exposing the columnar and row-wise structure of the document.

To suppress noise, border artefacts, and fine font variations, the profiles are smoothed using a median filter:

$$\widetilde{P}(x) = \operatorname{med}\big\{P(x + i) \mid i \in [-\lfloor k/2 \rfloor, \lfloor k/2 \rfloor]\big\}. \tag{6}$$

The window sizes are set to $k_v = 201$ and $k_h = 101$ for vertical and horizontal profiles, respectively. A larger vertical window bridges inter-line gaps while preserving column boundaries.

From the smoothed profile, we define the *valley set* for a threshold $\tau \in (0, 1)$:

$$\mathcal{L}_\tau = \big\{x : \widetilde{P}(x) < \tau \max_x \widetilde{P}(x)\big\}. \tag{7}$$

Maximal connected intervals $[a_i, b_i] \subseteq \mathcal{L}_\tau$ with length $b_i - a_i \geq L_{\min}$ determine the cut centres

$$c_i = \left\lfloor \frac{a_i + b_i}{2} \right\rfloor. \tag{8}$$

The fixed parameters are $\tau = 0.12$, $L^v_{\min} = 40$ px, and $L^h_{\min} = 30$ px. An example of the profiles and detected valleys is shown in Fig. 3.

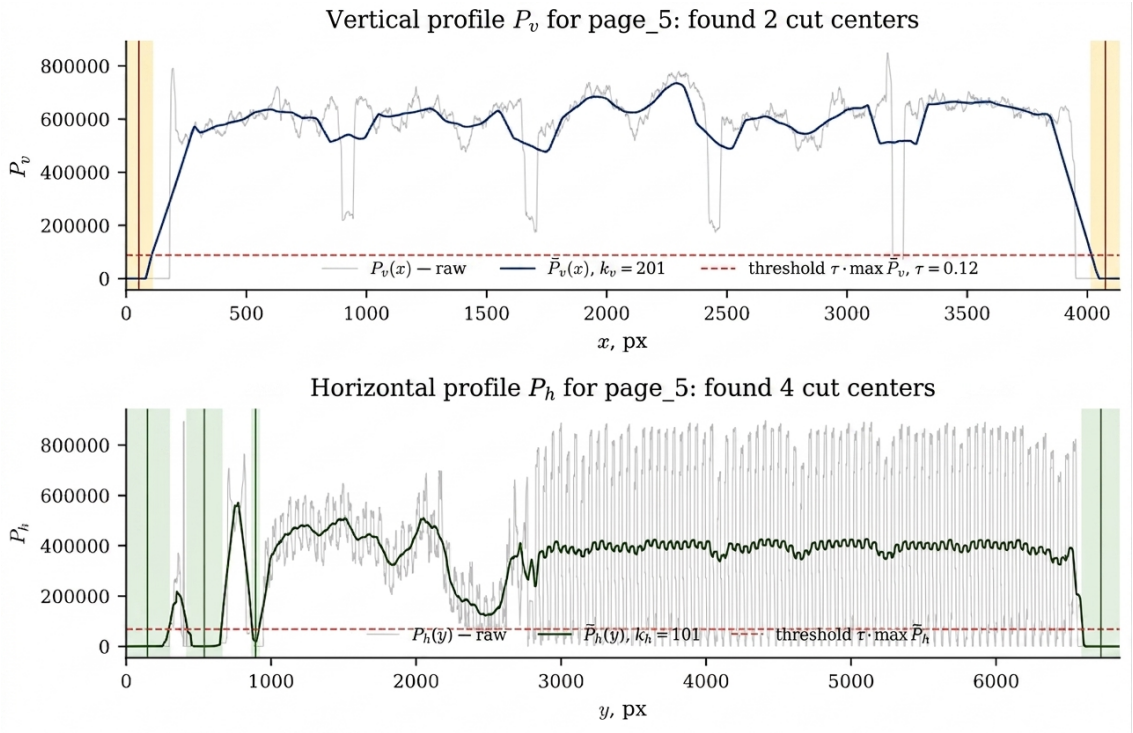

Figure 3: Vertical (P_v, top) and horizontal (P_h, bottom) projection profiles for a representative newspaper page.

Using the cut centres $\{c^v_i\}$ and $\{c^h_j\}$, we construct the extended sets $\mathcal{X} = \{0\} \cup \{c^v_i\} \cup \{W\}$ and $\mathcal{Y} = \{0\} \cup \{c^h_j\} \cup \{H\}$, sorted in ascending order. This yields a Cartesian partition of the page:

$$\mathcal{B} = \big\{ [x_i, x_{i+1}] \times [y_j, y_{j+1}] : x_i, x_{i+1} \in \mathcal{X},\ y_j, y_{j+1} \in \mathcal{Y} \big\}. \tag{9}$$

Each block $\beta = (x, y, w, h) \in \mathcal{B}$ is evaluated using its area $A(\beta) = wh$ and fill ratio

$$\rho(\beta) = \frac{1}{wh} \sum_{(u,v) \in \beta} B(u, v). \tag{10}$$

A block is retained if it satisfies the disjunctive condition

$$A(\beta) \geq A_{\min} \ \vee \ \rho(\beta) > \rho_{\min}, \quad A_{\min} = 3000 \text{ px}, \ \rho_{\min} = 10^{-3}. \tag{11}$$

This preserves small dense regions (e.g., logos) while removing large empty areas.

An additional area-ratio constraint is applied:

$$\frac{A(\beta)}{WH} \geq r_{\text{crop}}, \qquad r_{\text{crop}} = 0.05. \tag{12}$$

This step filters out insignificant regions, retaining only semantically meaningful large zones. The two-stage filtering effect is illustrated in Fig. 4.

Figure 4: Two-stage block filtering: (a) Candidate regions satisfying (11); (b) Remaining regions after applying (12).

The complete set of X-Cut++ hyperparameters θ is summarised in Table 1. All values are fixed and are not tuned per page.

3.3 Cascade decomposition

After the filter (12) the page is broken into a set of large crops $\{\beta_k\}$. Each crop undergoes three checks: vertical, horizontal, and a final vertical – forming the cascade

$$\beta_k \xrightarrow{\Pi_V} \{\beta_k^{(p)}\}_p \xrightarrow{\Pi_H} \{\beta_k^{(p,q)}\}_{p,q} \xrightarrow{\Pi'_V} \{\beta_k^{(p,q,r)}\}_{p,q,r}. \tag{13}$$

We now detail each step, following the logic of Algorithm 1.

Table 1: X-Cut++ configuration θ.

Parameter	Value	Role
Binarisation and morphology		
b	25	adaptive threshold window
C	10	threshold offset
K	15×3	dilation kernel
Projections		
k_v, k_h	201, 101	smoothing windows
τ	0.12	valley threshold
$L^v_{\min}, L^h_{\min}$	40, 30 px	minimum valley length
Block filters		
$A_{\min}$	3000 px	minimum area
$\rho_{\min}$	10^{-3}	fill ratio threshold
r_{crop}	0.05	area ratio

Detection of vertical separators. Inside a fragment β of height h and width w, an auxiliary Otsu binarisation $B^* = \text{Otsu}(I_\beta)$ is performed. Let $V(x) = \sum_y \mathbb{1}[B^*(y,x) > 0]$. A column is considered to contain a significant vertical separator if there exists a column x^* such that

$$\frac{V(x^*)}{h} \geq \alpha, \qquad \alpha = 0.75, \tag{14}$$

and, additionally, x^* belongs to a strong cluster: $S = \{x : V(x) > \beta_v \max_x V(x)\}$ with $\beta_v = 0.6$. A vertical split is made if at least one maximal connected interval of S contains an x^* satisfying (14).

Hough fallback. If the projection detector fails, the Canny edge detector [21] and the progressive probabilistic Hough transform [22] are invoked: HoughLinesP($\rho = 1, \theta = \pi/180, \text{thr} = 50$). A line $(x_1, y_1) \to (x_2, y_2)$ is accepted if

$$|\theta| > 80^\circ \ \wedge \ \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \geq \alpha h, \tag{15}$$

where $\theta = \arctan\big((y_2 - y_1)/(x_2 - x_1)\big)$.

Colour-aware rescue branch. A newspaper page often contains coloured panels (call-outs, rubrics, info blocks). Let $c_0 =$ `#e8f4f7` be the reference colour of the rubric. In HSV space a matching set is defined:

$$\Omega_{\text{HSV}} = \big\{p \in \Omega : |H(p) - H_0| \leq dH \pmod{180},\ |S(p) - S_0| \leq dS,\ |V(p) - V_0| \leq dV\big\}, \tag{16}$$

with $(dH, dS, dV) = (8, 60, 60)$. An alternative RGB criterion based on Euclidean distance,

$$\Omega_{\text{RGB}} = \{p \in \Omega : d(p) \leq \tau_{\text{RGB}}\},\ d(p) = \sqrt{(R(p) - R_0)^2 + (G(p) - G_0)^2 + (B(p) - B_0)^2}, \tag{17}$$

$\tau_{\text{RGB}} = 30$, is significantly less stable (see Section 4).

Morphological closing and opening with an elliptical kernel of radius 7 are applied to Ω_{HSV}; then connected components are extracted [23]. Only components whose area fraction

$\pi(\beta_c) \geq 15\%$ are retained. For a detected panel $\beta_c = (x_c, y_c, w_c, h_c)$ a virtual vertical separator is placed with a horizontal offset $\Delta = 40$ px from the corresponding edge:

$$x_{\text{line}} = \begin{cases} x_c - \Delta, & \text{left side,} \\ x_c + w_c + \Delta, & \text{right side.} \end{cases} \tag{18}$$

This line acts as a fallback cut when ordinary projections and Hough fail. In the current implementation the reference colour is set manually; however, it can be automatically extracted from a characteristic layout element (e.g., the top-of-page panel) when analysing several issues of the same newspaper.

Detection of horizontal separators. After vertical splitting, each part $\beta_k^{(p)}$ is searched for a horizontal line. Similarly to (14), let $U(y) = \sum_x \mathbb{1}[B^*(y, x) > 0]$. A line is deemed significant if

$$\frac{U(y^*)}{w} \geq \gamma, \qquad \gamma = 0.9, \tag{19}$$

and y^* lies outside a margin strip of width $\eta_{\text{edge}} = 0.05 \cdot h$ to exclude decorative borders.

Final vertical re-check and area-based assembly. Each temporary part $\beta_k^{(p,q)}$ is once more examined for vertical lines with the same threshold (14). This step catches vertical splits that become visible only after a horizontal cut (e.g., a block with a photo in the upper half and three text columns below). A final fragment $\beta_k^{(p,q,r)}$ is kept only if it satisfies the area-ratio condition

$$\frac{A\big(\beta_k^{(p,q,r)}\big)}{A\big(\beta_k^{(p,q)}\big)} \geq r_{\text{part}}, \qquad r_{\text{part}} = 0.15. \tag{20}$$

3.4 Post-OCR structural extraction

Tesseract 5 (language models `kaz+eng`) is applied to each final fragment. The output is a set of word rectangles with coordinates and confidence values $\{(t_i, x_i, y_i, w_i, h_i, \text{conf}_i)\}$. Words with $\text{conf}_i \leq 0$ or empty t_i are discarded.

The words are grouped into lines by the rule

$$\ell_i = \ell_{i-1} + \mathbb{1}\big[y_i - y_{i-1} > \delta_{\text{line}}\big], \quad \delta_{\text{line}} = 20 \text{ px}. \tag{21}$$

Within a line, words are sorted by x and concatenated with a space; for each line the mean height $\bar{h}_\ell$, minimal y_ℓ, and minimal x_ℓ are computed.

Title extraction relies on the relative height of lines. Let $L = \{\ell_1, \ldots, \ell_n\}$ be the set of lines of a fragment. The set of *title candidates* is

$$L_T = \big\{\ell \in L : \bar{h}_\ell \geq \theta_T \max_{\ell'} \bar{h}_{\ell'}\big\}, \quad \theta_T = 0.7. \tag{22}$$

Candidates are sorted by y_ℓ and concatenated into the title as long as the vertical gap between successive lines does not exceed $G_T = 150$ px:

$$\ell_{k+1} \in \text{Title} \iff y_{\ell_{k+1}} - y_{\ell_k} \leq G_T. \tag{23}$$

The abstract is extracted immediately below the title. Starting from the first body line after the title, with a minimum offset $\Delta_{\text{post}} = 50$ px, lines are grouped into columns by the left coordinate x_ℓ (greedy procedure, threshold $\sigma_{\text{col}} = 200$ px). In the first (leftmost) column a continuous sequence of lines is collected until the vertical gap exceeds $G_A = 70$ px. The obtained lines are normalised (hyphenation joins) and form the `abstract` field.

An author line is identified by a formal predicate. Let t be the text of a line, $|t|$ its character length, $w(t)$ the number of words. Then

$$\text{IsAuthor}(t) = \mathbb{1}[w(t) \in [2,3]] \cdot \mathbb{1}[|t| \le 45] \cdot \prod_{w \in t} \phi(w), \tag{24}$$

where

$$\phi(w) = \mathbb{1}\big[(w_0 \in U \wedge w_{1:} \in L^*) \vee w \in U^*\big], \tag{25}$$

U – uppercase letters, L – lowercase letters. Words must be either of the form "Capital + lowercase*" or fully capitalised (surname).

The body is assembled by traversing columns left to right. For a given reference line $\ell^* = (t^*, x^*, y^*, h^*)$ we define

$$\text{NextBelow}(\ell^*) = \arg\min_{\ell \in L'} (y_\ell - y_b^*), \tag{26}$$

where $L' = \{\ell : y_\ell > y_b^*,\ |x_\ell - x^*| < \lambda,\ |\bar{h}_\ell - h^*| < \eta,\ y_\ell - y_b^* < G_{\max}\}$ with $\lambda = 100$ px, $\eta = 10$ px, $G_{\max} = 500$ px. After exhausting one column, the traversal moves to the next expected left coordinate $x^* + \sigma_{\text{col}}$. The newspaper name is filtered out using the fuzzy matching condition

$$\text{partial\_ratio}(t, \text{«Egemen Qazaqstan»}) \ge 70, \tag{27}$$

where partial_ratio is the Levenshtein-based metric [24] implemented in RapidFuzz [25]. The body is normalised by removing hyphenation breaks and collapsing multiple spaces.

An auxiliary rule-based scoring classifier assigns each OCR element to one of the classes $\{\text{title}, \text{abstract}, \text{body}\}$. For an element e with bounding box (x_1, y_1, x_2, y_2) and text t_e, a 7-dimensional feature vector $\mathbf{f}_e$ is computed (normalised y-centre, normalised height, number of words, fraction of uppercase letters, number of sentences, punctuation coefficient, presence of keywords). The score for class c is a sum of indicator contributions:

$$\text{score}_c(e) = \sum_{j=1}^{m_c} a_j^{(c)}\, \mathbb{1}\big[\mathbf{f}_e \in R_j^{(c)}\big], \tag{28}$$

and the predicted class is $\hat{c}(e) = \arg\max_c \text{score}_c(e)$. The complete list of indicator regions and weights is given in Table 2.

4 Experimental results

4.1 Dataset and experimental setup

The dataset comprises **five issues** of the national Kazakh newspaper *Egemen Qazaqstan* published between 22 January and 3 February 2024. The PDF files were converted to PNG

Table 2: Indicator contributions for the scoring rule (28).

Class	Indicator $R_j^{(c)}$	Weight $a_j^{(c)}$
T (title)	$y_e^c < 0.15$	$+2.0$
T (title)	$0.15 \le y_e^c < 0.30$	$+0.8$
T (title)	$h_e^r > 0.08$	$+1.0$
T (title)	$w_e^c \le 12 \wedge \kappa_e > 0.35$	$+2.0$
A (abstract)	keyword present	$+3.0$
A (abstract)	$s_e \in [1,3] \wedge w_e^c \le 120$	$+1.5$
A (abstract)	π_e (punctuation coefficient)	$+1.0\,\pi_e$
B (body)	$w_e^c/200$ (long text)	$+1.0\,\frac{w_e^c}{200}$
B (body)	default fallback if total score $= 0$	$+0.5$

at 300 DPI using PyMuPDF (`fitz`) [26], yielding 80 page images, of which 72 contained editorial content (excluding full-page advertisements). All images have the identical resolution 4134×6851 px. The OCR engine is Tesseract 5 with models `kaz+eng`; post-processing is implemented in Python with pandas / RapidFuzz. All pipeline parameters are fixed to the values declared in Sections 3–4 and were kept identical across all issues. Because the pages come from digital master copies, no significant skew is present, and a separate deskew step was not required.

4.2 Comprehensive evaluation of segmentation and structural extraction

This subsection consolidates the quantitative segmentation results, colour space comparison, ablation study, and structural extraction assessment.

Quantitative segmentation. Table 3 summarises the cascade for three representative pages. Across the entire dataset (72 pages) the X-Cut++ pipeline produced **230 article-level fragments** (mean 3.19 fragments per page). The minimum was 2 fragments on simple pages, the maximum 11 on a heavily multi-column page. All fragments were successfully OCR-ed and saved as synchronised `*.png + *.txt` pairs.

Table 3: Cascade segmentation: number of objects at each stage.

Page	Raw blocks	After (11)	After (12)	After $\Pi_V \circ \Pi_H$	Final $\lvert\mathcal{F}\rvert$
Evaluation Page A	12	12	2	4	4
Evaluation Page B	18	18	4	8	10
Evaluation Page C	—	—	3	—	3

Colour space comparison. The HSV and RGB colour matchers were evaluated on a representative newspaper page (Fig. 5, Table 4). The HSV criterion remains stable, as the Hue channel is largely invariant to brightness and saturation variations in printed offset. In contrast, the Euclidean RGB distance between the reference colour $(232, 244, 247)$ and paper white $(255, 255, 255)$ equals $\sqrt{23^2 + 11^2 + 8^2} \approx 26.5 < 30$, causing the background to be absorbed. As a result, the RGB mask fails to isolate coloured panels.

Figure 5: Colour-aware segmentation on a representative newspaper page. (a) Original page. (b) HSV mask with $(dH, dS, dV) = (8, 60, 60)$. (c) RGB mask with $\tau_{\mathrm{RGB}} = 30$.

Table 4: HSV vs. RGB mask coverage.

Method	# components	Coverage, %	Outcome	Suitability
HSV	5–10 small	~ 1.1	localised	suitable
RGB	1 macro	~ 81.7	merges background	degraded

Table 5: Ablation study: number of fragments on evaluation pages.

Configuration	Page A	Page B
Projection only	2	6
+ Hough fallback	3	8
+ HSV (full X-Cut++)	4	10

Ablation study and sensitivity. We evaluate the contribution of each rescue branch by disabling components and measuring the number of final fragments on two challenging pages (Table 5). The full X-Cut++ recovers the correct segmentation, whereas the projection-only variant misses a vertical split inside a coloured panel.

Sensitivity analysis over $\tau \in [0.08, 0.16]$ shows that the number of fragments varies by at most one, with cut positions remaining stable (shift < 10 px). Varying $\alpha \in [0.65, 0.85]$ does not introduce false splits, confirming the robustness of the selected hyperparameters.

Structural extraction from OCR. Fifteen fragments from different issues were processed by the structural parser. All titles (15/15) and abstracts (15/15) were correctly extracted, along with 11/15 author lines; the remaining cases correspond to layouts where author information is absent.

On a manually annotated newspaper article from the evaluation dataset, the scoring classifier (28) correctly classified all seven OCR elements (title, subtitle, abstract fragment, author, two text blocks, continuation label).

5 Discussion

The experimental evaluation shows that X-Cut++ reliably decomposes complex newspaper pages into article-level units. A cascade of projection cuts, augmented with Hough and HSV rescue branches, resolves cases where pure projection analysis fails, in particular coloured panels that disrupt Otsu binarisation.

The transparency of each processing step, together with a fixed hyperparameter set, ensures fully deterministic and reproducible segmentation. This property is critical for large-scale archival digitisation, where auditability and traceability are required.

The structural post-OCR parser reconstructs the canonical article structure (title, abstract, author, body) with high accuracy, indicating that rule-based methods are sufficient for the regular editorial layout of *Egemen Qazaqstan*. Extracted semantic units directly support downstream applications such as indexing, summarisation, and integration with large language model pipelines. In particular, the structured output provides a natural interface for LLM-based agents performing tabular and textual interpretation, as explored for Kazakh-language data in [19].

Several limitations point to directions for future work. First, the current parameters are tuned for the *Egemen Qazaqstan* layout; adapting X-Cut++ to other newspapers with different typography requires automatic calibration, potentially based on a small set of annotated exemplars. Second, the colour-aware branch depends on a predefined reference colour; estimating dominant rubric colours automatically would improve generalisation. Third, the structural parser assumes a relatively fixed article structure; more diverse layouts may require adaptive heuristics or lightweight learned models trained on X-Cut++ outputs.

Fourth, the pipeline assumes orthogonal, non-skewed pages; a robust deskew preprocessing stage is required for scanned archives. Finally, OCR quality is limited by Tesseract performance on Kazakh; integrating a stronger domain-adapted model would further improve recognition accuracy and downstream parsing. Addressing these issues is part of ongoing work aimed at scaling the pipeline to long-term newspaper archives while preserving interpretability and low-resource efficiency.

6 Conclusion

We have proposed, formalised, and empirically validated X-Cut++—a hybrid, fully interpretable layout-aware pipeline for the structural decomposition of Kazakh-language newspaper pages. The method combines adaptive binarisation, smoothed projection profiles, morphological operations, HSV colour segmentation, probabilistic Hough line detection, and a rule-based post-OCR parser.

The mathematical core is a cascade of one-dimensional projection cuts with geometric filtering. On a dataset of five issues of *Egemen Qazaqstan* (72 editorial pages), X-Cut++ produced 230 article-level fragments. Structural evaluation on 15 test fragments yielded perfect title and abstract extraction, and correct author detection in all cases where an author was present.

HSV-based colour segmentation was found to be significantly more stable than a Euclidean RGB criterion. X-Cut++ requires no labelled training data for the layout task, provides deterministic and reproducible results, and remains fully auditable.

Future work will focus on automatic hyperparameter tuning, integration of advanced OCR engines, and scaling to yearly newspaper archives.

Acknowledgements

The authors thank the editorial staff of *Egemen Qazaqstan* for making the electronic issues available for research, and the developer communities of Tesseract, OpenCV, PyMuPDF, and RapidFuzz for their freely accessible tools.

Funding

This research was funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan grant number BR24993001 “Creation of a large language model (LLM) to maintain the implementation of Kazakh language and increase the technological progress”.

References

[1] Smith, R. (2007). An overview of the Tesseract OCR engine. In *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR)*, 2, 629–633.

[2] Breuel, T. M. (2002). Two geometric algorithms for layout analysis. In *Document Analysis Systems V (LNCS 2423)*, 188–199. Springer.

[3] Huang, Y., Lv, T., Cui, L., Lu, Y., & Wei, F. (2022). LayoutLMv3: Pre-training for document AI with unified text and image masking. In *Proceedings of ACM Multimedia*, 4083–4091.

[4] Kim, G., Hong, T., Yim, M., Nam, J., Park, J., Yim, J., Hwang, W., Yun, S., Han, D., & Park, S. (2022). OCR-free document understanding transformer. In *Proceedings of ECCV*, 498–517.

[5] Nagy, G. (2000). Twenty years of document image analysis in PAMI. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), 38–62.

[6] O’Gorman, L. (1993). The document spectrum for page layout analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11), 1162–1173.

[7] Nagy, G., Seth, S., & Viswanathan, M. (1992). A prototype document image analysis system for technical journals. *Computer*, 25(7), 10–22.

[8] Wong, K. Y., Casey, R. G., & Wahl, F. M. (1982). Document analysis system. *IBM Journal of Research and Development*, 26(6), 647–656.

[9] Kise, K., Sato, A., & Iwata, M. (1998). Segmentation of page images using the area Voronoi diagram. *Computer Vision and Image Understanding*, 70(3), 370–382.

[10] Zhong, X., Tang, J., & Jimeno Yepes, A. (2019). PubLayNet: Largest dataset ever for document layout analysis. In *Proceedings of ICDAR*, 1015–1022.

[11] Pfitzmann, B., Auer, C., Dolfi, M., Nassar, A. S., & Staar, P. (2022). DocLayNet: A large human-annotated dataset for document-layout analysis. In *Proceedings of KDD*, 3743–3751.

[12] Mukhamediev, A., Yermekbayev, B., et al. (2022). Layout analysis of Kazakh newspapers using YOLO-based detectors. *Bulletin of L.N. Gumilyov ENU. Mathematics, Computer Science, Mechanics Series*, 3, 45–58.

[13] Clausner, C., Antonacopoulos, A., & Pletschacher, S. (2015). ENP image and ground truth dataset of historical newspapers. In *Proceedings of ICDAR*, 931–935.

[14] Kaspari, A., Bostanbekov, K., Tolegenov, A., et al. (2019). End-to-end OCR system for the Kazakh language with a focus on press archives. *Bulletin of the Karaganda University. Mathematics Series*, 94(2), 78–92.

[15] Smith, R. (2013). History of the Tesseract OCR engine: What worked and what didn't. In *Proceedings of SPIE 8658: Document Recognition and Retrieval XX*.

[16] Du, Y., Li, C., Guo, R., Yin, X., Liu, W., Zhou, J., Bai, Y., Yu, Z., Yang, Y., Dang, Q., & Wang, H. (2020). PP-OCR: A practical ultra lightweight OCR system. *arXiv preprint arXiv:2009.09941*.

[17] Nugumanova, A. B., Apayev, K. S., Baiburin, Y. M., Mansurova, M., & Ospan, A. (2022). QURMA: A table extraction pipeline for knowledge base population. *Bulletin of KazNU. Mathematics, Mechanics, Computer Science Series*, 114(2). https://doi.org/10.26577/JMMCS.2022.v114.i2.08

[18] Bauyrzhan, K., Mansurova, M., & Ospan, A. (2023). Fine-tuning the Wav2Vec2 model for Kazakh speech: A study on a limited corpus. In *Proceedings of IEEE SIST 2023*. https://doi.org/10.1109/SIST58284.2023.10223504

[19] Ospan, A., Mussa, A., Mansurova, M., & Sarsembayeva, T. (2025). LLM agents for enhanced tabular data interpretation: A perspective. In *Proceedings of IEEE SIST 2025*. https://doi.org/10.1109/SIST61657.2025.11139242

[20] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66.

[21] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), 679–698.

[22] Matas, J., Galambos, C., & Kittler, J. (2000). Robust detection of lines using the progressive probabilistic Hough transform. *Computer Vision and Image Understanding*, 78(1), 119–137.

[23] He, L., Chao, Y., Suzuki, K., & Wu, K. (2009). Fast connected-component labeling. *Pattern Recognition*, 42(9), 1977–1987.

[24] Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8), 707–710.

[25] Bachmann, M. RapidFuzz: Rapid fuzzy string matching in Python and C++. Available at https://github.com/maxbachmann/RapidFuzz (accessed 2024).

[26] Artifex Software. PyMuPDF (fitz): Python bindings for MuPDF. Available at https://pymupdf.readthedocs.io (accessed 2024).

Information about authors:

Assel Ospan – Doctor of Philosophy (PhD) in System Engineering, Senior Lecturer at the Department of Artificial Intelligence and Big Data, Al-Farabi Kazakh National University (Almaty, Kazakhstan, e-mail: assel.ospan@kaznu.edu.kz, ORCID: 0000-0002-1860-6997).

Madina Mansurova – Candidate of Physical and Mathematical Sciences, Professor, Dean of the Faculty of Mechanics and Mathematics, Al-Farabi Kazakh National University (Almaty, Kazakhstan, e-mail: madina.mansurova@kaznu.edu.kz, ORCID: 0000-0002-9680-2758).

Aisha Sailau – Researcher at Al-Farabi Kazakh National University (Almaty, Kazakhstan, e-mail: aishasailau3@gmail.com, ORCID: 0009-0003-8251-3327).

Talshyn Sarsembayeva – Doctor of Philosophy (PhD) in AI in medicine, Senior Lecturer and Deputy Head for Research and Innovation at the Department of Artificial Intelligence and Big Data, Al-Farabi Kazakh National University (Almaty, Kazakhstan, e-mail: talshyn.sagdatbek@kaznu.edu.kz, ORCID: 0000-0001-7668-2640).

Авторлар туралы мәлімет:

Оспан Әсел Ғалымжанқызы – жүйелік инженерия мамандығы бойынша философия докторы (PhD), әл-Фараби атындағы Қазақ ұлттық университетінің Жасанды интеллект және Big Data кафедрасының аға оқытушысы (Алматы, Қазақстан, электрондық пошта: assel.ospan@kaznu.edu.kz, ORCID: 0000-0002-1860-6997).

Мансурова Мадина Есімханқызы – физика-математика ғылымдарының кандидаты, профессор, әл-Фараби атындағы Қазақ ұлттық университетінің механика-математика факультетінің деканы (Алматы, Қазақстан, электрондық пошта: madina.mansurova@kaznu.edu.kz, ORCID: 0000-0002-9680-2758).

Сайлау Айша – әл-Фараби атындағы Қазақ ұлттық университетінің Жасанды интеллект және Big Data кафедрасының ғылыми қызметкері (Алматы, Қазақстан, электрондық пошта: aishasailau3@gmail.com, ORCID: 0009-0003-8251-3327).

Сәрсембаева Талшын Сағдатбекқызы – медицинадағы жасанды интеллект мамандығы бойынша философия докторы (PhD),әл-Фараби атындағы Қазақ ұлттық университетінің

Жасанды интеллект және Big Data кафедрасының аға оқытушысы, ғылыми-инновациялық қызмет жөніндегі кафедра меңгерушісінің орынбасары (Алматы, Қазақстан, электрондық пошта: talshyn.sagdatbek@kaznu.edu.kz, ORCID: 0000-0001-7668-2640).

Информация об авторах:

Оспан Асель Галымжановна – доктор философии (PhD) по специальности «Системная инженерия», старший преподаватель кафедры искусственного интеллекта и Big Data Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: assel.ospan@kaznu.edu.kz, ORCID: 0000-0002-1860-6997).

Мансурова Мадина Есимхановна – кандидат физико-математических наук, профессор, декан механико-математического факультета Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: madina.mansurova@kaznu.edu.kz, ORCID: 0000-0002-9680-2758).

Сайлау Айша – научный сотрудник кафедры искусственного интеллекта и Big Data Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: aishasailau3@gmail.com, ORCID: 0009-0003-8251-3327).

Сарсембаева Талшын Сагдатбековна – доктор философии (PhD) по специальности "ИИ в медицине старший преподаватель кафедры искусственного интеллекта и Big Data, заместитель заведующего кафедрой по научно-инновационной деятельности Казахского национального университета имени аль-Фараби (Алматы, Казахстан, электронная почта: talshyn.sagdatbek@kaznu.edu.kz, ORCID: 0000-0001-7668-2640).

Received: May 06, 2026
Accepted: June 18, 2026