УДК 004.932

РАЗРАБОТКА МОДУЛЯ ДЕТЕКТИРОВАНИЯ ЭЛЕМЕНТОВ ИНТЕРФЕЙСА ДЛЯ ПОВЫШЕНИЯ ДОСТУПНОСТИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

Васильев Д. Е. (ИТМО)

Научный руководитель — к. п. н, доцент факультета ПИиКТ, Государев И. Б. (ИТМО)

Введение. По данным Всемирной организации здравоохранения, не менее 2,2 миллиарда человек в мире имеют нарушения зрения [1]. Основным инструментом взаимодействия таких пользователей с мобильными устройствами служат скринридеры — в частности, VoiceOver на платформе iOS. Скринридер озвучивает пользователю элементы интерфейса, опираясь на семантическую разметку, заложенную разработчиком: accessibility-метки, hints и роли элементов. Однако при использовании кастомных UI-компонентов или нестандартных графических элементов эта разметка зачастую отсутствует, что делает приложение полностью недоступным.

Масштабный анализ мобильных приложений показал, что более 77% из них содержат критические нарушения доступности, напрямую связанные с отсутствием текстовых меток у графических элементов [2]. При этом подавляющее большинство существующих решений требуют вмешательства разработчика на этапе создания приложения и не способны устранить проблему ретроактивно [3]. Задача автоматической генерации семантической разметки на основе визуального представления экрана была поставлена и частично решена в работах Apple и Google, однако предложенные подходы либо закрыты для сторонней разработки, либо не охватывают весь спектр нозологий и нестандартных UI-паттернов [4]. В соответствии с требованиями ГОСТ Р 52872-2019 мобильные приложения обязаны обеспечивать доступность для пользователей с ограниченными возможностями здоровья [5], что создаёт как нормативные, так и технические предпосылки для разработки открытого автоматизированного решения.

Основная часть. Предлагаемый подход предусматривает создание промежуточного программного модуля, встраиваемого между мобильным приложением и системой VoiceOver. Модуль перехватывает рендер экрана и с помощью модели компьютерного зрения автоматически детектирует элементы пользовательского интерфейса, определяет их тип и функцию, а затем генерирует семантические метки, которые передаются скринридеру вместо отсутствующей нативной разметки. Такой подход принципиально не требует доступа к исходному коду приложения, что позволяет повысить доступность уже существующих продуктов без участия их разработчиков.

В качестве обучающих данных используется датасет Rico [6], содержащий 72 219 UI-экранов из 9 772 приложений с аннотациями view hierarchies и 25 категориями UI-компонентов. В основу детектора положена архитектура нейронной сети с одноэтапным детектором типа YOLO, дообученная на размеченных мобильных интерфейсах. Для генерации текстовых описаний элементов применяется мультимодальный подход: визуальный энкодер на базе ResNet извлекает признаки изображения компонента, языковой декодер на базе Transformer синтезирует

описание на естественном языке. Данный подход показал BLEU-4 = 0,254 и CIDEr = 1,31 на датасете Widget Captioning, содержащем более 162 000 аннотаций для 61 285 UI-элементов [7].

Архитектура модуля включает три последовательных этапа: (1) детектирование и классификация элементов интерфейса по снимку экрана; (2) генерация семантических текстовых описаний на основе мультимодального энкодера-декодера; (3) инъекция сформированных меток в дерево доступности, передаваемое VoiceOver через Accessibility API. В рамках проектирования прототипа определены контрольные метрики качества: mAP детектирования UI-элементов и BLEU/CIDEr для оценки релевантности сгенерированных описаний. Планируется формирование собственного валидационного датасета с аннотациями, верифицированными незрячими пользователями.

Проведённый анализ существующих решений показал, что наиболее близкой работой является Screen Recognition (Apple, CHI 2021), реализованная в iOS 14 в виде закрытого on-device модуля [4]. Предлагаемый модуль отличается открытостью архитектуры, возможностью дообучения на узкоспециализированных UI-фреймворках и потенциальной кроссплатформенностью.

Выводы. В ходе исследования выявлена ключевая причина недоступности мобильных приложений для незрячих пользователей — отсутствие семантической разметки у кастомных элементов интерфейса, не поддерживаемых стандартным Accessibility API. Предложена и обоснована концепция промежуточного модуля детектирования на основе компьютерного зрения, создающего недостающий семантический слой между приложением и скринридером VoiceOver.

Научная новизна работы состоит в формировании сквозного конвейера автоматической генерации accessibility-метаданных из пиксельного представления экрана без доступа к исходному коду приложения. Разработанная концепция позволяет ретроактивно повышать доступность уже опубликованных приложений, что принципиально отличает её от существующих подходов, требующих участия разработчика.

В перспективе планируется реализация прототипа и его тестирование с привлечением незрячих пользователей, расширение функционала для поддержки нарушений моторики, а также адаптация модуля под платформу Android. Результаты работы могут быть применены для аудита доступности и автоматического исправления нарушений в экосистеме мобильных приложений.

Список использованных источников:

1. World Health Organization. World Report on Vision. — Geneva : WHO, 2019. — URL: https://www.who.int/publications-detail-redirect/world-report-on-vision (дата обращения: 10.02.2025).
2. Ross A. S. An Epidemiology-Inspired Large-Scale Analysis of Android App Accessibility / A. S. Ross, X. Zhang, J. Fogarty, J. O. Wobbrock // ACM Transactions on Accessible Computing. — 2020. — Vol. 13, No. 1. — pp. 1–36. — DOI: 10.1145/3348797.
3. Chen J. Unblind Your Apps: Predicting Natural-Language Labels for Mobile GUI Components by Deep Learning / J. Chen, C. Chen, Z. Xing, X. Xu, L. Zhu, G. Li, J. Wang // Proceedings of the 42nd International Conference on Software Engineering (ICSE '20). — 2020. — pp. 322–334. — DOI: 10.1145/3377811.3380327.
4. Zhang X. Screen Recognition: Creating Accessibility Metadata for Mobile Applications from Pixels / X. Zhang, L. de Greef, A. Swearngin [et al.] // Proceedings of the ACM CHI

Conference on Human Factors in Computing Systems (CHI '21). — 2021. — Article 275. — DOI: 10.1145/3411764.3445186.
5. ГОСТ Р 52872-2019. Интернет-ресурсы и другая информация, представленная в электронно-цифровой форме. Требования доступности для людей с инвалидностью. — М. : Росстандарт, 2019. — Введён в действие 01.04.2020. — URL: http://docs.cntd.ru/document/1200167693 (дата обращения: 10.02.2025).
6. Deka B. Rico: A Mobile App Dataset for Building Data-Driven Design Applications / B. Deka, Z. Huang, C. Franzen [et al.] // Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17). — 2017. — pp. 845–854. — DOI: 10.1145/3126594.3126651.
7. Li Y. Widget Captioning: Generating Natural Language Description for Mobile User Interface Elements / Y. Li, G. Li, L. He [et al.] // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP '20). — 2020. — pp. 5495–5510. — DOI: 10.18653/v1/2020.emnlp-main.443.