

PaddleOCR 3.0 Technical Report

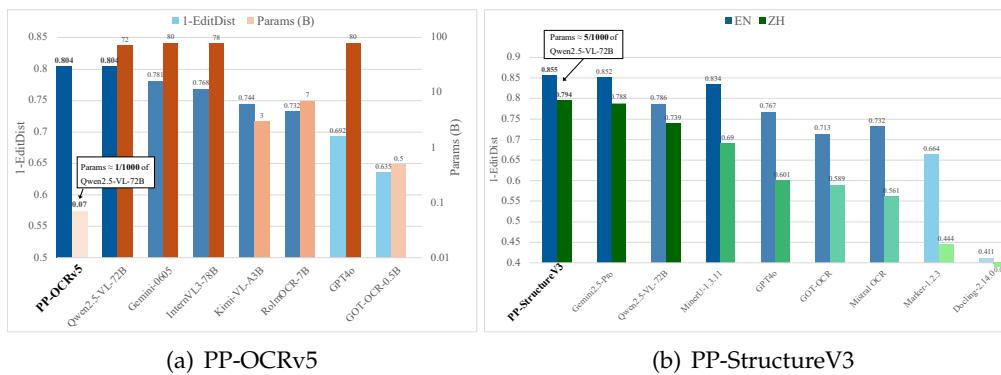
Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, Yue Zhang, Wenyu Lv, Kui Huang, Yichao Zhang, Jing Zhang, Jun Zhang, Yi Liu, Dianhai Yu, Yanjun Ma

PaddlePaddle Team, Baidu Inc.
paddleocr@baidu.com

- ⌚ Source Code: <https://github.com/PaddlePaddle/PaddleOCR>
- 📄 Document: <https://paddlepaddle.github.io/PaddleOCR>
- 🤗 Models & Online Demo: <https://huggingface.co/PaddlePaddle>

Abstract

This technical report introduces PaddleOCR 3.0, an Apache-licensed open-source toolkit for OCR and document parsing. To address the growing demand for document understanding in the era of large language models, PaddleOCR 3.0 presents three major solutions: (1) PP-OCRv5 for multilingual text recognition, (2) PP-StructureV3 for hierarchical document parsing, and (3) PP-ChatOCRv4 for key information extraction. Compared to mainstream vision-language models (VLMs), these models with fewer than 100 million parameters achieve competitive accuracy and efficiency, rivaling billion-parameter VLMs. In addition to offering a high-quality OCR model library, PaddleOCR 3.0 provides efficient tools for training, inference, and deployment, supports heterogeneous hardware acceleration, and enables developers to easily build intelligent document applications.



(a) PP-OCRv5

(b) PP-StructureV3

Figure 1 | Performance comparison of PP-OCRv5 and PP-StructureV3 with their respective counterparts. The evaluation set for PP-OCRv5 is our self-built dataset, which includes multiple writing formats such as Simplified Chinese, Traditional Chinese, Chinese Pinyin, English, and Japanese. The evaluation set for PP-StructureV3 is OmniDocBench (Ouyang et al., 2025). The term "1-EditDist" refers to 1 – Edit Distance, where a higher value indicates better performance.

PaddleOCR 3.0 Teknik Raporu

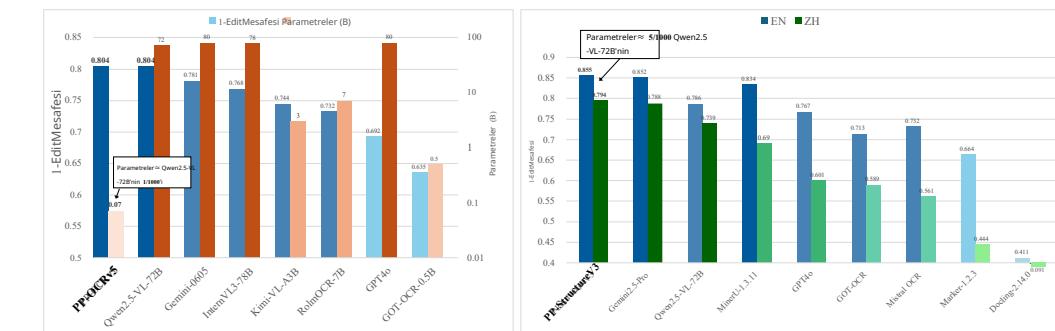
Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, Yue Zhang, Wenyu Lv, Kui Huang, Yichao Zhang, Jing Zhang, Jun Zhang, Yi Liu, Dianhai Yu, Yanjun Ma

PaddlePaddle Takımı, Baidu Inc.
paddleocr@baidu.com

- ⌚ Kaynak Kodu: <https://github.com/PaddlePaddle/PaddleOCR>
- 📄 Belge: <https://paddlepaddle.github.io/PaddleOCR>
- 🤗 Modeller ve Çevrimiçi Demo: <https://huggingface.co/PaddlePaddle>

Özet

Bu teknik rapor, OCR ve belge ayrıştırma için Apache lisanslı açık kaynaklı bir araç takımı olan PaddleOCR 3.0'ı tanıtmaktadır. Büyük dil modelleri çağında belge anlaması talebinin artmasıyla başa çıkmak için PaddleOCR 3.0 üç ana çözüm sunar : (1) Çok dilli metin tanıma için PP-OCRv5, (2) hiyerarşik belge ayrıştırma için PP-StructureV3 ve (3) anahtar bilgi çıkarımı için PP-ChatOCRv4. Ana akım görme-dil modelleri (VLM'ler) ile karşılaşıldığında, 100 milyondan az parametreye sahip bu modeller, milyarlarca parametreli VLM'lere rakip olacak düzeyde rekabetçi doğruluk ve verimlilik elde etmektedir. Yüksek kaliteli bir OCR model kütüphanesi sunmanın yanı sıra, PaddleOCR 3.0 eğitim, çıkışım ve dağıtım için verimli araçlar sağlar, heterojen donanım hızlandırmayı destekler ve geliştiricilerin akıllı belge uygulamalarını kolayca oluşturmasını olanak tanır.



(a) PP-OCRv5

(b) PP-StructureV3

Şekil 1 | PP-OCRv5 ve PP-StructureV3'ün ilgili muadilleriyle performans karşılaştırması. PP-OCRv5 için değerlendirme seti, Basitleştirilmiş Çince, Geleneksel Çince, Çince Pinyin, İngilizce ve Japonca gibi birden fazla yazı formatını içeren, kendi oluşturduğumuz veri setimizdir. PP-StructureV3 için değerlendirme seti OmniDocBench'tir (Ouyang vd., 2025). "1-EditDist" terimi, daha yüksek bir değerin daha iyi performansı gösterdiği 1 – Düzenleme Mesafesi anlamına gelir.

1. Introduction

Optical Character Recognition (OCR) is a foundational technology that enables the conversion of images containing text, scanned documents into structured, machine-readable text. Its significance has never been more pronounced than in the current era of artificial intelligence, where massive volumes of unstructured visual data are generated and consumed daily across scientific, industrial, and social domains. The recent surge in large language models (LLMs)(Achiam et al., 2023; Baidu-ERNIE-Team, 2025; Guo et al., 2025; Yang et al., 2025) and Retrieval-Augmented Generation (RAG) systems(Lewis et al., 2020) has further elevated the strategic importance of OCR: it is no longer sufficient for OCR systems to merely transcribe text accurately—they must now serve as critical enablers in the construction of high-quality datasets, facilitate knowledge extraction, and act as bridges between the visual and semantic layers of modern AI systems.

The evolution of OCR technology reflects the broader trajectory of computer vision and natural language processing. Early OCR systems(Casey and Lecolinet, 1996; Mori et al., 1999), based on hand-crafted features and rule-based heuristics, performed adequately under controlled conditions but quickly reached their limits when confronted with the complexity and diversity of real-world scenarios. The advent of deep learning, particularly convolutional neural networks (CNNs) and their derivatives, ushered in a new era of data-driven OCR, enabling substantial improvements in recognition accuracy, robustness, and adaptability(Goodfellow et al., 2014; Shi et al., 2015). However, as large-scale AI applications proliferate, new requirements have emerged: OCR engines need to handle a broader range of documents—from handwritten notes and multilingual content to rare or historical scripts, and complex layouts with tables, charts, and embedded images. Furthermore, in industrial and research settings alike, OCR is increasingly expected to support downstream tasks such as document understanding, key information extraction (KIE), and semantic search, often as part of end-to-end intelligent workflows.

In recent years, the rapid advancement of LLMs and RAG systems has fundamentally transformed the landscape of information retrieval and knowledge management. These systems rely heavily on the availability of high-quality, diverse, and accurately labeled textual corpora for both pre-training and inference. OCR, in this context, is not simply a data acquisition tool, but a linchpin technology that fuels the entire pipeline—from digitizing vast archives of scientific literature to enabling real-time question answering over heterogeneous document collections. The accuracy and comprehensiveness of OCR outputs directly influence the performance and trustworthiness of LLM-based applications, especially in domains where information is predominantly shared in scanned or image-based formats (e.g., legal documents, historical records, scientific papers, and business forms). Moreover, RAG architectures, which combine retrieval mechanisms with generative modeling, are particularly sensitive to the quality of underlying document representations. Inadequate OCR can propagate errors, introduce noise, or omit critical content, thereby undermining the effectiveness of retrieval and the factual correctness of generated responses.

Despite these pressing needs, existing OCR solutions still face significant challenges in practical deployment. Traditional pipelines typically struggle with low-quality scans, complex backgrounds, non-standard fonts, and multi-modal documents that blend text with figures, tables, or handwritten annotations. The diversity of real-world languages, scripts, and writing styles further complicates the recognition process, requiring not only robust visual modeling but also powerful language understanding capabilities. In addition, industrial and research users increasingly demand lightweight, easily deployable solutions that can be adapted to different hardware constraints and integrated seamlessly with larger AI ecosystems. The open-source community has played a pivotal role in democratizing access to advanced OCR technology,

1. Giriş

Optik Karakter Tanıma (OCR), metin içeren görüntüler ve taranmış belgeleri yapılandırılmış, makine tarafından okunabilir metne dönüştüren temel bir teknolojidir. Önemi, bilimsel, endüstriyel ve sosyal alanlarda her gün büyük hacimli yapılandırılmamış görsel verinin üretildiği ve tüketildiği günümüz yapay zeka çağında hiç bu kadar belirgin olmamıştır. Büyük dil modellerindeki (LLM'ler) (Achiam ve diğerleri, 2023; Baidu-ERNIE-Team, 2025; Guo ve diğerleri, 2025; Yang ve diğerleri, 2025) ve Geri Getirme Artırılmış Üretim (RAG) sistemlerindeki (Lewis ve diğerleri, 2020) son dönemdeki yükseliş, OCR'nin stratejik önemini daha da artırmıştır: Artık OCR sistemlerinin yalnızca metni doğru bir şekilde deşifre etmesi yeterli değildir; yüksek kaliteli veri kümelerinin oluşturulmasında kritik kolaylaştırıcılar olarak hizmet etmeleri, bilgi çıkarımı kolaylaştırmaları ve modern yapay zeka sistemlerinin görsel ve semantik katmanları arasında köprü görevi görmeleri gerekmektedir.

OCR teknolojisinin evrimi, bilgisayar görüşü ve doğal dil işlemenin genel seyrini yansıtmaktadır. El yapımı özelliklere ve kural tabanlı sezgisel yöntemlere dayanan erken OCR sistemleri (Casey ve Lecolinet, 1996; Mori ve diğerleri, 1999), kontrollü koşullar altında yeterli performans göstermiş ancak gerçek dünya senaryolarının karmaşaklısı ve çeşitliliği ile karşılaşlıklarında hızla sınırlarına ulaşmışlardır. Derin öğrenmenin, özellikle de evrimsel sınır ağlarının (CNN'ler) ve türevlerinin ortaya çıkışını, veri odaklı OCR'in yeni bir çağını başlatarak, tanıma doğruluğu, sağlamlığı ve uyarlanabilirliğinden önemli gelişmeler sağlamıştır (Goodfellow ve diğerleri, 2014; Shi ve diğerleri, 2015). Ancak, büyük ölçekli yapay zeka uygulamaları yaygınlaşıkça, yeni gereksinimler ortaya çıkmıştır: OCR motorlarının, el yazısı notlardan ve çok dilli içerikten nadir veya tarihi yazılıara ve tablolar, grafikler ve gömülü görüntüler içeren karmaşık düzenlere kadar daha geniş bir belge yelpazesini işlemesi gerekmektedir. Ayrıca, hem endüstriyel hem de araştırma ortamlarında, OCR'dan, genellikle uçtan uca akıllı iş akışlarının bir parçası olarak, belge anlama, anahtar bilgi çıkarımı (KIE) ve anlamsal arama gibi sonraki görevleri desteklemesi giderek daha fazla beklenmektedir.

Son yıllarda, LLM'ler ve RAG sistemlerinin hızla ilerlemesi, bilgi erişimi ve bilgi yönetimi alanını temelden dönüştürmüştür. Bu sistemler, hem ön eğitim hem de çıkarım için yüksek kaliteli, çeşitli ve doğru etiketlenmiş metinsel veri kümelerinin bulunabilirliğine büyük ölçüde bağımlıdır. OCR, bu bağlamda, yalnızca bir veri toplama aracı değil, geniş bilimsel literatür arşivlerini dijitalleştirerek heterojen belge koleksiyonları üzerinde gerçek zamanlı soru yanıtlamayı sağlamaya kadar tüm süreci besleyen temel bir teknolojidir.

OCR çıktılarının doğruluğu ve kapsamlılığı, özellikle bilgilerin ağırlıklı olarak taranmış veya görüntü tabanlı formatta paylaşıldığı alanlarda (örneğin, yasal belgeler, tarihi kayıtlar, bilimsel makaleler ve iş formları) LLM tabanlı uygulamaların performansını ve güvenilirliğini doğrudan etkiler. Dahası, geri alma mekanizmalarını üretken modellemeyle birleştiren RAG mimarileri, temel belge temsillerinin kalitesine özellikle duyarlıdır. Yetersiz OCR, hataları yayabilir, gürültü ekleyebilir veya kritik içeriği atlayabilir; böylece geri almanın etkinliğini ve üretilen yanıtların olgusal doğruluğunu zayıflatır.

Bu acil ihtiyaçlara rağmen, mevcut Optik Karakter Tanıma (OCR) çözümleri pratik dağıtımda hala önemli zorluklarla karşılaşmaktadır. Geleneksel işlem hatları genellikle düşük kaliteli taramalar, karmaşık arka planlar, standart olmayan yazı tipleri ve metni şekiller, tablolar veya el yazısı notları birleştiren çok modlu belgelerle mücadele etmektedir. Gerçek dünya dillerinin, alfabelerinin ve yazı stillerinin çeşitliliği, tanıma sürecini daha da karmaşık hale getirmekte; bu da sadece sağlam görsel modelleme değil, aynı zamanda güçlü dil anlama yetenekleri de gerektirmektedir. Ek olarak, endüstriyel ve araştırma kullanıcıları, farklı donanım kısıtlamalarına uyarlanabilecek ve daha büyük yapay zeka ekosistemleriyle sorunsuz bir şekilde entegre edilebilen, hafif ve kolayca dağıtolabilen çözümler talep etmektedir. Açık kaynak topluluğu, gelişmiş OCR teknolojisine erişimi demokratikleştirmede önemli bir rol oynamıştır,

yet there remains a gap between academic research prototypes and production-ready systems capable of supporting the stringent requirements of dataset construction, RAG workflows, and large-scale document intelligence.

PaddleOCR 1.x & 2.x: Advancements and Innovations in Open-Source OCR Technology

PaddleOCR has emerged as a prominent open-source project addressing these multifaceted challenges. Since its initial release in 2020, PaddleOCR has adhered to the principles of comprehensive coverage, end-to-end workflow, and lightweight efficiency, setting new standards for both usability and technical excellence in the OCR domain. Anchored by the PP-OCR series, PaddleOCR has evolved through multiple iterations—each pushing the boundaries of text detection, recognition, and document analysis. Early versions such as PP-OCRV1(Du et al., 2020) focused on achieving an optimal balance between accuracy and speed, making OCR accessible for resource-constrained environments. Subsequent releases (PP-OCRV2(Du et al., 2021), v3(Li et al., 2022b), and v4) incrementally improved recognition performance, extended language coverage, and introduced sophisticated models for handwriting and rare character recognition. A notable advancement has been the integration of document structural understanding via the PP-Structure series, enabling PaddleOCR to move beyond text lines and paragraphs to address complex layout analysis, table structure recognition (e.g., SLANet(Li et al., 2022a)), and other advanced parsing tasks. These capabilities have made PaddleOCR a critical engine for automated document processing, intelligent archiving, information extraction, and, increasingly, for supporting the data pipelines of LLMs and RAG systems.

The adoption and impact of PaddleOCR in both academic and industrial communities are evidenced by its widespread use and vibrant developer ecosystem. With more than 50,000 stars on GitHub as of June 2025, and its deployment as the core OCR engine in projects such as MinerU(Wang et al., 2024), RAGFlow(KevinHuSh, 2023), and UmiOCR(hiroi sora, 2022), PaddleOCR has become an indispensable tool for digitization initiatives, knowledge management platforms, and AI-driven document analysis workflows. Notably, PaddleOCR has played a central role in the construction of high-quality document datasets for large model training, enabling researchers to assemble diverse, accurately annotated corpora spanning multiple languages, domains, and document types. Its modular architecture and rich API ecosystem facilitate seamless integration with RAG pipelines, where efficient and accurate OCR is essential for document ingestion, retrieval indexing, and context provision to generative models.

As PaddleOCR's user base has expanded, so has the range of feedback and requirements from the community. Users have highlighted persistent needs in areas such as robust handwriting recognition, improved support for multi-language and rare script recognition, more powerful document parsing for complex layouts, and advanced key information extraction. These demands are further amplified by the growing scale and dynamism of LLM and RAG applications, where the ability to extract, structure, and semantically interpret information from diverse documents is a prerequisite for building reliable, responsive, and intelligent systems. Aware of these trends and our responsibility as a leading open-source platform, we remain committed to continuously improving PaddleOCR to meet the evolving challenges of the field.

PaddleOCR 3.0: A New Milestone in Enhancing Text Recognition and Document Parsing

In this context, we introduce PaddleOCR 3.0, a major release designed to systematically enhance text recognition accuracy and document parsing capabilities, with a particular focus on the complex scenarios encountered in modern AI applications. PaddleOCR 3.0 encompasses several core innovations. First, it presents the high-precision text recognition pipeline PP-OCRV5, which leverages advanced model architectures and training strategies to deliver state-of-the-

ancak akademik araştırma prototipleri ile veri seti oluşturma, RAG iş akışları ve büyük ölçekli belge zekasının katı gereksinimlerini destekleyebilen üretime hazır sistemler arasında hala bir boşluk bulunmaktadır.

PaddleOCR 1.x & 2.x: Açık Kaynak OCR Teknolojisinde Gelişmeler ve Yenilikler PaddleOCR

, bu çok yönlü zorlukları ele alan önemli bir açık kaynak projesi olarak ortaya çıkmıştır. 2020'deki ilk sürümünden bu yana PaddleOCR, kapsamlı kapsama alanı, uçtan uca iş akışı ve hafif verimlilik ilkelerine sadık kalarak, OCR alanında hem kullanılabılırlik hem de teknik mükemmellik için yeni standartlar belirlemiştir. PP-OCR serisi tarafından desteklenen PaddleOCR, her biri metin tespiti, tanıma ve belge analizinin sınırlarını zorlayan birçok iterasyonla evrimleşmiştir. PP-OCRV1 (Du vd., 2020) gibi erken sürümler, doğruluk ve hız arasında en uygun dengeyi sağlamaya odaklanarak, kaynak kısıtlı ortamlar için OCR'ı erişilebilir kılmıştır. Sonraki sürümler (PP-OCRV2 (Du vd., 2021), v3 (Li vd., 2022b) ve v4), tanıma performansını kademeli olarak iyileştirmiştir, dil kapsamını genişletmiş ve el yazısı ile nadir karakter tanıma için gelişmiş modeller sunmuştur.

Dikkate değer bir ilerleme, PP-Structure serisi aracılığıyla belge yapısal anlayışının entegrasyonu olmuştur; bu sayede PaddleOCR, metin satırları ve paragraflarının ötesine geçerek karmaşık düzen analizi, tablo yapısı tanıma (örn. SLANet (Li vd., 2022a)) ve diğer gelişmiş ayırtırma görevlerini ele alabilmiştir. Bu yetenekler, PaddleOCR'ı otomatik belge işleme, akıllı arşivleme, bilgi çıkarma ve giderek artan bir şekilde LLM'ler ile RAG sistemlerinin veri süreçlerini desteklemek için kritik bir motor haline getirmiştir.

PaddleOCR'ın hem akademik hem de endüstriyel topluluklardaki benimsenmesi ve etkisi, yaygın kullanımı ve canlı geliştirici ekosistemi ile kanıtlanmaktadır. Haziran 2025 itibarıyla GitHub'da 50.000'den fazla yıldız sahib olması ve MinerU (Wang vd., 2024), RAGFlow (KevinHuSh, 2023) ve UmiOCR (hiroi sora, 2022) gibi projelerde temel OCR motoru olarak dağıtılmışıyla PaddleOCR, dijitalleşme girişimleri, bilgi yönetimi platformları ve yapay zeka odaklı belge analiz iş akışları için vazgeçilmez bir araç haline gelmiştir. Özellikle, PaddleOCR, büyük model eğitimi için yüksek kaliteli belge veri kümelerinin oluşturulmasında merkezi bir rol oynamış, araştırmacıların birden fazla dil, alan ve belge türünü kapsayan çeşitli ve doğru şekilde açıklanmış derlemler oluşturmasını sağlamıştır. Modüler mimarisi ve zengin API ekosistemi, verimli ve doğru OCR'nin belge alımı, geri çağrıma indekslemesi ve üretken modellere bağlam sağlanması için esas olduğu RAG ardışık düzenleri ile sorunsuz entegrasyonu kolaylaştırmaktadır.

PaddleOCR'ın kullanıcı tabanı genişledikçe, topluluktan gelen geri bildirim ve gereksinim yelpazesi de çeşitlenmiştir. Kullanıcılar, sağlam el yazısı tanıma, çok dilli ve nadir komut dosyası tanıma için geliştirilmiş destek, karmaşık düzenler için daha güçlü belge ayırtırma ve gelişmiş anahtar bilgi çıkarımı gibi alanlarda devam eden ihtiyaçları vurgulamışlardır. Bu talepler, LLM ve RAG uygulamalarının artan ölçeği ve dinamizmiyle daha da güçlenmektedir; zira çeşitli belgelerden bilgi çıkarma, yapılandırma ve anlamsal olarak yorumlama yeteneği, güvenilir, duyarlı ve akıllı sistemler oluşturmak için bir ön koşuldur. Bu eğilimlerin ve onde gelen bir açık kaynak platformu olarak sorumluluğumuzun bilincinde olarak, alanın gelişen zorluklarını karşılamak üzere PaddleOCR'ı sürekli geliştirmeye kararlıyız.

PaddleOCR 3.0: Metin Tanıma ve Belge Ayırtırmayı Gelişirmede Yeni Bir Dönüm Noktası

Bu bağlamda, metin tanıma doğruluğunu ve belge ayırtırma yeteneklerini sistematik olarak geliştirmek amacıyla tasarlanmış büyük bir sürüm olan PaddleOCR 3.0'ı tanıtıyoruz; özellikle modern yapay zeka uygulamalarında karşılaşılan karmaşık senaryolara odaklanılmıştır. PaddleOCR 3.0, birkaç temel yeniliği bünyesinde barındırmaktadır. İlk olarak, gelişmiş model mimarilerini ve eğitim stratejilerini kullanarak en son teknoloji ürünü yüksek hassasiyetli metin tanıma hattı PP-OCRV5'i sunmaktadır.

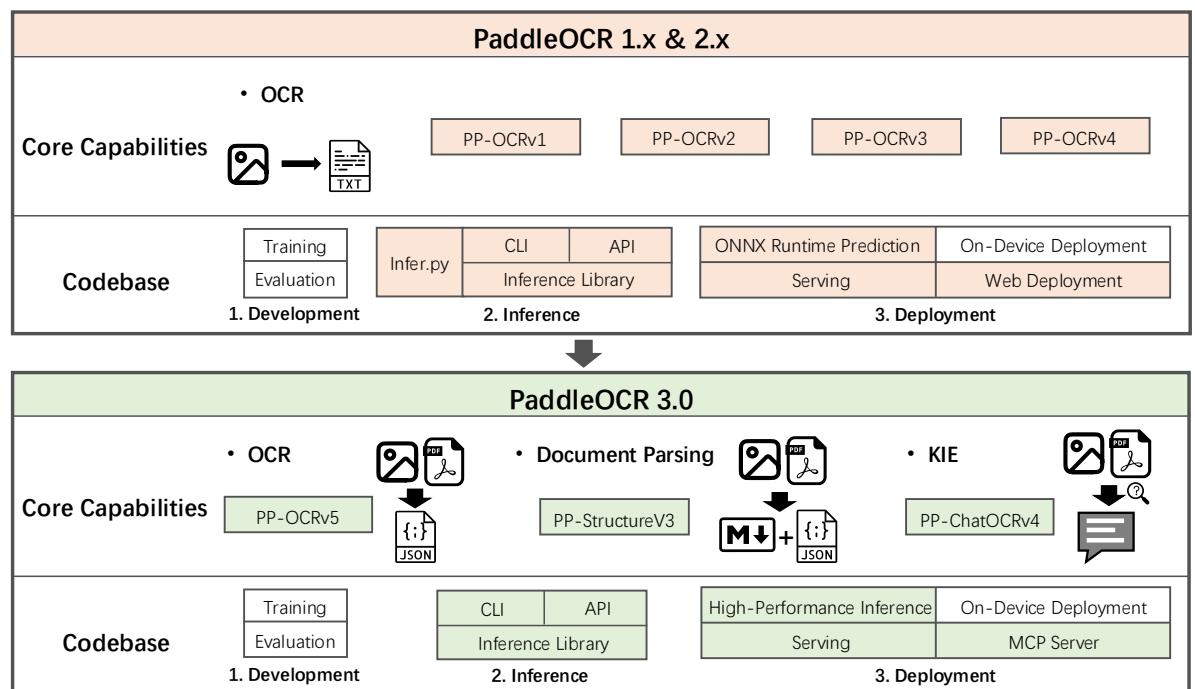
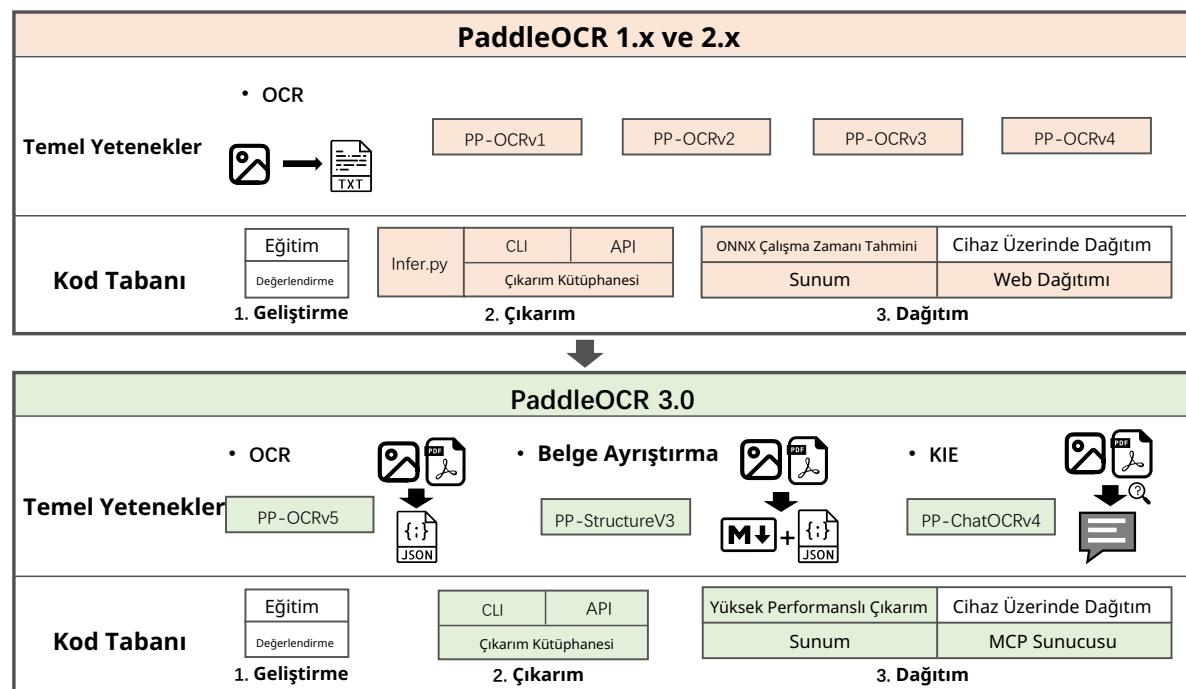


Figure 2 | Evolution from PaddleOCR 1.x & 2.x to PaddleOCR 3.0. Different colors have been employed to denote areas with notable discrepancies between PaddleOCR 1.x & 2.x and PaddleOCR 3.0.

art accuracy across printed, handwritten, and multilingual documents, while maintaining efficiency suitable for both cloud and edge deployment. Moreover, PP-OCRv5 achieves unified recognition of Simplified Chinese, Traditional Chinese, Chinese Pinyin, English, and Japanese within a single model. Second, PaddleOCR 3.0 includes PP-StructureV3, a document parsing solution that integrates layout analysis, table recognition, and structure extraction in an end-to-end framework, enabling accurate and scalable document understanding for forms, invoices, scientific literature, and more. Third, recognizing the need for deeper semantic integration, we introduce PP-ChatOCRv4, a system that combines lightweight OCR models with large language models to facilitate key information extraction, context-aware question answering, and flexible document comprehension—capabilities that are essential for powering RAG pipelines and intelligent document agents. In addition, PaddleOCR 3.0 extends its coverage with dedicated solutions for specialized tasks such as seal text recognition, formula recognition, and chart analysis, further expanding its utility for both research and industrial use cases.

The contributions of PaddleOCR 3.0 are not limited to technical innovation; the release continues to prioritize openness, usability, and extensibility, with a robust API ecosystem, comprehensive model zoo, and active community support. In this version, several legacy design flaws have been revised to provide cleaner and extensible API and CLI, while maintaining a reasonable degree of backward compatibility. At the deployment level, PaddleOCR 3.0 has been restructured to deliver a more streamlined, out-of-the-box experience and improved integration capabilities with LLMs. By targeting key challenges in complex OCR scenarios and aligning with the fundamental needs of data construction for LLMs and RAG pipelines, PaddleOCR 3.0 aspires to serve as an efficient, intelligent, and open infrastructure for document AI. We hope that this advancement will accelerate the development of intelligent automation and knowledge-driven AI systems, foster new research and application frontiers at the intersection of vision



Şekil 2 | PaddleOCR 1.x ve 2.x'ten PaddleOCR 3.0'a Evrim. PaddleOCR 1.x ve 2.x ile PaddleOCR 3.0 arasındaki belirgin farklılıklarını belirtmek için farklı renkler kullanılmıştır. PaddleOCR 3.0.

basılı, el yazısı ve çok dilli belgelerde yüksek doğruluk sağlarken, hem bulut hem de kenar dağıtımları için uygun verimliliği korumaktadır. Ayrıca, PP-OCRv5, Basitleştirilmiş Çince, Genel Çince, Çince Pinyin, İngilizce ve Japonca'nın tek bir model içinde birleşik tanımmasını sağlar. İkinci olarak, PaddleOCR 3.0, formlar, faturalar, bilimsel literatür ve daha fazlası için doğru ve ölçeklenebilir belge anlamayı sağlayan, düzen analizi, tablo tanıma ve yapı çıkarımı uça bir çerçevede birleştiren bir belge ayrıştırma çözümü olan PP-StructureV3'ü içerir. Üçüncü, daha derin semantik entegrasyon ihtiyacını anlayarak, RAG ardışık düzenleri ve akıllı belge araçları için temel olan anahtar bilgi çıkarma, bağlantı duyarlı soru yanıtlama ve esnek belge anlamayı yeteneklerini kolaylaştırmak üzere hafif OCR modellerini büyük dil modelleriyle birleştirilen bir sistem olan PP-ChatOCRv4'ü sunuyoruz. Ayrıca, PaddleOCR 3.0, mühür metni tanıma, formül tanıma ve grafik analizi gibi özel görevler için ayrılmış çözümlerle kapsamını genişleterek, hem araştırma hem de endüstriyel kullanım senaryoları için faydasını daha da artırmaktadır.

PaddleOCR 3.0'ın katkıları teknik yenilikle sınırlı değildir; sürüm, sağlam bir API ekosistemi, kapsamlı bir model deposu ve aktif topluluk desteği ile açılığa, kullanılabilirliğe ve genişletilebilirliğe öncelik vermeye devam etmektedir. Bu sürümde, makul bir geriye dönük uyumluluk derecesini korurken, daha temiz ve genişletilebilir bir API ile CLI sağlamak amacıyla çeşitli eski tasarım kusurları gözden geçirilmiştir. Dağıtım seviyesinde, PaddleOCR 3.0, daha akıcı, kullanıcıya hazır bir deneyim ve LLM'lerle geliştirilmiş entegrasyon yetenekleri sunmak üzere yeniden yapılandırılmıştır. Karmaşık OCR senaryolarındaki temel zorlukları hedefleyerek ve LLM'ler ile RAG ardışık düzenleri için veri oluşturmanın temel ihtiyaçlarıyla uyum sağlayarak, PaddleOCR 3.0, belge yapay zekası için verimli, akıllı ve açık bir altyapı olarak hizmet etmeyi amaçlamaktadır. Bu gelişmenin, akıllı otomasyon ve bilgi odaklı yapay zeka sistemlerinin gelişimini hızlandıracığını, vizyon

and language, and promote document processing toward higher levels of intelligence and automation.

2. Core Capabilities

PaddleOCR 3.0 comprises three core capabilities: PP-OCRv5, PP-StructureV3 and PP-ChatOCRv4. This section elaborates on the problems addressed by these capabilities, details of the model solution, and their performance metrics.

2.1. PP-OCRv5

PP-OCRv5 is a high-precision and lightweight OCR system designed to perform effectively in a wide range of scenarios. It supports a diverse range of scripts within a single model, including Simplified Chinese, Traditional Chinese, Chinese Pinyin, English, and Japanese. To address the diverse hardware environments and varying requirements for inference speed, PP-OCRv5 offers two distinct model variants: a server version and a mobile version. The server version is specifically optimized for systems equipped with hardware accelerators such as GPUs, thereby enabling accelerated inference and higher throughput. In contrast, the mobile version is tailored for deployment in CPU-only environments, with optimizations targeting resource-constrained devices. Unless otherwise specified, all mentions of PP-OCRv5 in this paper refer by default to the server version. Figure 3 illustrates the framework of PP-OCRv5, which comprises four key components: image preprocessing, text detection, text line orientation classification, and text recognition. The following will introduce these four components.

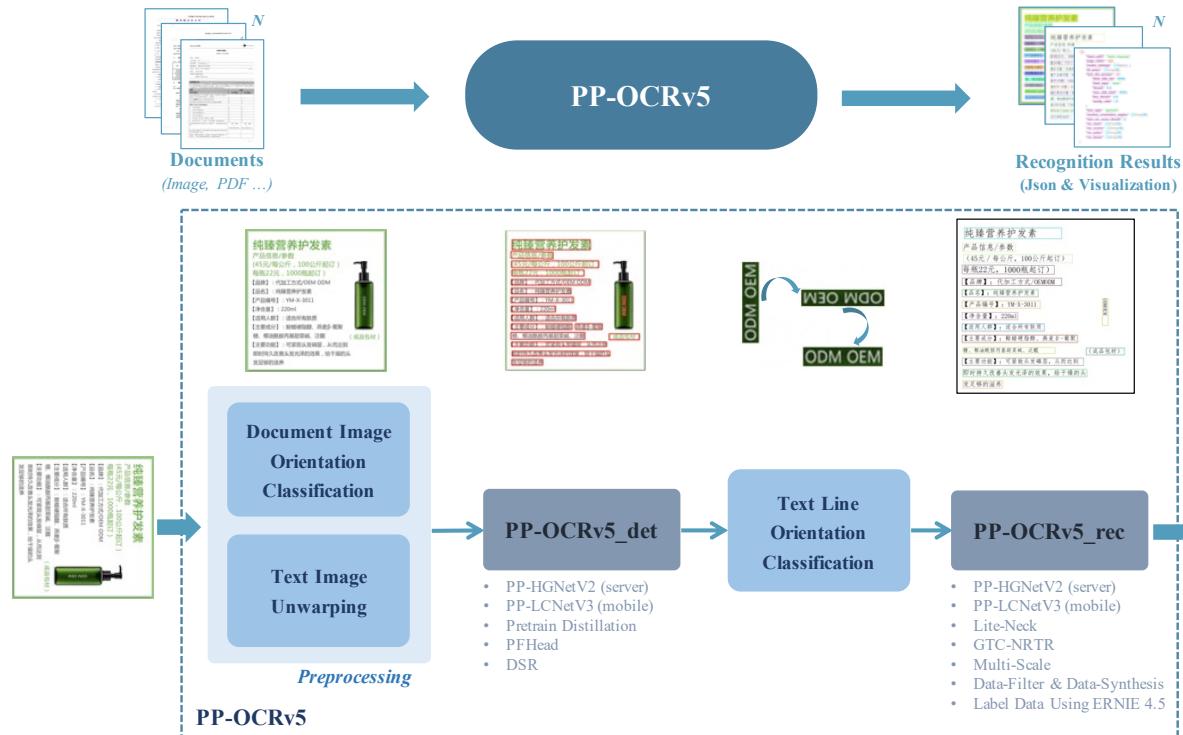


Figure 3 | Pipeline of PP-OCRv5. The pipeline includes image preprocessing, text region detection, text line orientation classification, and text recognition, ultimately extracting the text from images and outputting it as structured textual content.

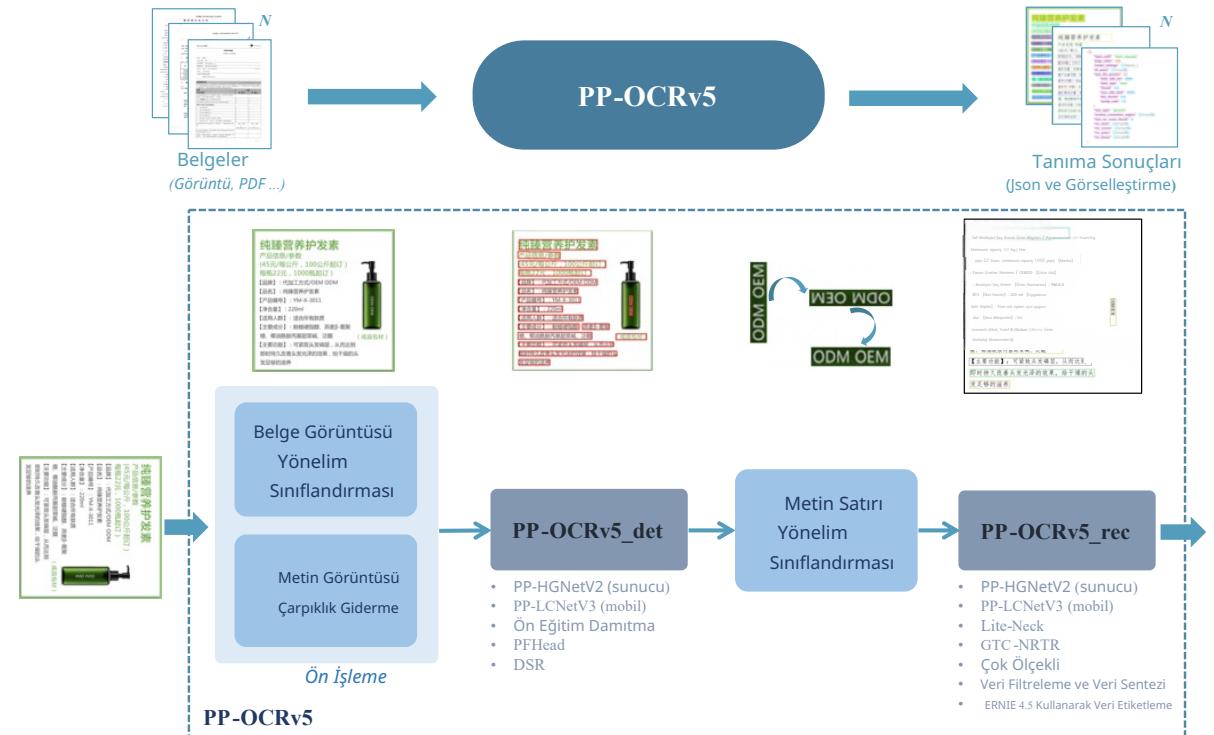
ve dil özellikleriyle, belge işlemeyi daha yüksek zeka ve otomasyon seviyelerine taşımayı hedefler.

2. Temel Yetkinlikler

PaddleOCR 3.0, PP-OCRv5, PP-StructureV3 ve PP-ChatOCRv4 olmak üzere üç temel yetkinliği bünyesinde barındırır. Bu bölümde, söz konusu yetkinliklerin ele aldığı problemler, model çözüm detayları ve performans metrikleri açıklanmaktadır.

2.1. PP-OCRv5

PP-OCRv5, geniş bir senaryo yelpazesinde etkili performans sergilemek üzere tasarlanmış, yüksek hassasiyetli ve hafif bir OCR sistemidir. Basitleştirilmiş Çince, Geleneksel Çince, Çince Pinyin, İngilizce ve Japonca dahil olmak üzere tek bir model içinde çeşitli yazı tiplerini destekler. Farklı donanım ortamlarına ve çıkışım hızı gereksinimlerine uyum sağlamak amacıyla PP-OCRv5, sunucu ve mobil versiyon olmak üzere iki farklı model varyantı sunar. Sunucu versiyonu, GPU gibi donanım hızlandırıcılarla donatılmış sistemler için özel olarak optimize edilmiştir; bu sayede hızlandırılmış çıkışım ve daha yüksek verimlilik sağlar. Buna karşılık, mobil sürüm, kaynak kısıtlı cihazları hedefleyen optimizasyonlarla yalnızca CPU ortamlarında dağıtım için özel olarak tasarlanmıştır. Aksi belirtildikçe, bu belgede PP-OCRv5'ten yapılan tüm atıflar varyantı olarak sunucu sürümünü ifade eder. Şekil 3, görüntü ön işleme, metin algılama, metin satırı yönelim sınıflandırması ve metin tanıma olmak üzere dört temel bileşenden oluşan PP-OCRv5 çerçevesini göstermektedir. Aşağıda bu dört bileşen tanıtılacaktır.



Şekil 3 | PP-OCRv5'in İş Akışı. İş akışı; görüntü ön işleme, metin bölgesi tespiti, metin satırı yönelim sınıflandırması ve metin tanıma işlemlerini içerir ve nihayetinde görüntülerden metni çıkararak yapılandırılmış metinsel içerik olarak çıktı verir.

1. Image Preprocessing Module: The image preprocessing module is crucial for preparing the input images by enhancing their quality and adjusting distortions or orientation issues. This process lays a solid foundation for accurate text detection and recognition. PP-OCRv5 includes an optional image preprocessing module to handle image rotation and geometric distortion. This module includes an image orientation classification model based on PP-LCNet (Cui et al., 2021) and a text image unwarping model built on UVDoc (Verhoeven et al., 2023). Users can choose to use these features based on their application scenarios.

2. Text Detection Model: The PP-OCRv5 text detection model enhances its predecessor, PP-OCRv4, through optimizations in three key aspects: network architecture, distillation strategy, and data augmentation. Firstly, PP-OCRv5 adopts the more advanced PP-HGNetV2¹ as its backbone network, replacing the previous PP-HGNet. Moreover, PP-OCRv5 enhances model robustness through knowledge distillation, utilizing a visual encoder from the advanced GOT-OCR2.0 (Wei et al., 2024) as the teacher model to transfer knowledge by aligning feature representations with the PP-HGNetV2 student. Finally, PP-OCRv5 detection model enhances text detection performance by incorporating advanced data augmentation techniques, including hard case mining via ERNIE-4.5-VL-424B-A47B comparison and text line-based multilingual strategies (random synthesis, rotation, blurring, geometric transformations). Notably, PP-OCRv5 detection model retains effective strategies from PP-OCRv4, such as the PFHead (Parallel Fusion Head) architecture and the DSR (Dynamic Scale-aware Refinement) training strategy. Overall, PP-OCRv5 improves the model's ability to generalize across diverse datasets, leading to more accurate and reliable text detection in real-world scenarios.

3. Text Line Orientation Classification Model: In PP-OCRv5, the text line orientation classification model is primarily responsible for determining and correcting the orientation of detected text lines. When the input text lines are misoriented (e.g., inverted or rotated), this model can automatically identify and rectify their direction to ensure that they are in the standard readable orientation. This step guarantees that the subsequent text recognition model receives a correctly oriented text line, thereby improving the overall accuracy and robustness of the OCR system.

4. Text Recognition Model: The PP-OCRv5 recognition model employs a dual-branch architecture with PP-HGNetV2 as the backbone: one branch (GTC-NRTR) uses attention-based training to enhance sequence modeling (Hu et al., 2020), while the other (SVTR-HGNet) focuses on efficient inference with CTC loss. During training, the GTC-NRTR branch guides the SVTR-HGNet branch (Du et al., 2022), but only the lightweight SVTR-HGNet branch is used for prediction, ensuring both accuracy and speed (Li et al., 2022c). For data construction, PP-OCRv5 combines traditional models with the ERNIE-4.5-VL-424B-A47B to automatically annotate and filter high-quality handwritten samples, including rare characters generated through synthesis. Additionally, large-scale labeled data is obtained from documents like PDFs and e-books using automated parsing and edit distance filtering. These data construction strategies also provide a solid data foundation for the overall performance improvement of PP-OCRv5.

Accordingly, the core contributions of PP-OCRv5 are as follows:

1. Unified Multilingual Modeling: PP-OCRv5 achieves unified recognition of Simplified Chinese, Traditional Chinese, Chinese Pinyin, English, and Japanese within a single model. Through an innovative unified architecture design, the model maintains a compact size under 100 MB. This resolves efficiency bottlenecks caused by integrating multiple models in multilingual scenarios, significantly simplifying industrial deployment processes. An example

¹https://github.com/PaddlePaddle/PaddleClas/blob/release/2.6/docs/en/models/PP-HGNetV2_en.md

1. Görüntü Ön İşleme Modülü : Görüntü ön işleme modülü, giriş görüntülerinin kalitesini artıracak ve bozulmaları veya yönelim sorunlarını düzelterek hazırlamak için kritik öneme sahiptir. Bu süreç, doğru metin algılama ve tanıma için sağlam bir temel oluşturur. PP-OCRv5, görüntü döndürme ve geometrik bozulmaları ele almak için istege bağlı bir görüntü ön işleme modülü içerir. Bu modül, PP-LCNet (Cui vd., 2021) tabanlı bir görüntü yönelim sınıflandırma modeli ile UVDoc (Verhoeven vd., 2023) üzerine inşa edilmiş bir metin görüntüsü bozulma düzeltme modelini barındırır. Kullanıcılar, uygulama senaryolarına bağlı olarak bu özellikleri kullanmayı tercih edebilirler.

2. Metin Algılama Modeli : PP-OCRv5 metin algılama modeli, ağ mimarisi, damıtma stratejisi ve veri artırma olmak üzere üç temel alandaki optimizasyonlarla selefı PP-OCRv4'ü geliştirir. İlk olarak, PP-OCRv5, önceki PP-HGNet'in yerine daha gelişmiş PP-HGNetV2¹ 'yi temel ağ olarak benimser. Ayrıca, PP-OCRv5, gelişmiş GOT-OCR2.0'dan (Wei et al., 2024) gelen görsel bir kodlayıcıyı öğretmen model olarak kullanarak, PP-HGNetV2 öğrencisi ile özellik temsillerini hizalayarak bilgi aktarımı yoluyla model sağlamlığını artırır. Son olarak, PP-OCRv5 algılama modeli, ERNIE-4.5-VL-424B-A47B karşılaşması aracılığıyla zorlu durum madenciliği ve metin satırı tabanlı çok dilli stratejiler (rastgele sentez, döndürme, bulanıklaştırma, geometrik dönüşümler) dahil olmak üzere gelişmiş veri artırma tekniklerini dahil ederek metin algılama performansını iyileştirir. Özellikle, PP-OCRv5 algılama modeli, PFHead (Paralel Füzyon Başlığı) mimarisi ve DSR (Dinamik Ölçek Duyarlı İyileştirme) eğitim stratejisi gibi PP-OCRv4'ten bilinen etkili stratejileri korur. Genel olarak, PP-OCRv5 , modelin çeşitli veri kümeleri arasında genelleme yeteneğini geliştirmek gerçek dünya senaryolarında daha doğru ve güvenilir metin algılamasına yol açmaktadır.

3. Metin Satırı Yönü Sınıflandırma Modeli : PP-OCRv5'te metin satırı yönü sınıflandırma modeli, algılanan metin satırlarının yönünü belirlemekten ve düzeltmekten birincil olarak sorumludur. Giriş metin satırları yanlış yönlendirilmişse (örn. ters çevrilmiş veya döndürülmüş), bu model yönlerini otomatik olarak tanımlayabilir ve düzeltebilir, böylece standart okunabilir yönde olmalarını sağlar. Bu adım, sonraki metin tanıma modelinin doğru yönlendirilmiş bir metin satırı olmasını garanti ederek OCR sistemi'nin genel doğruluğunu ve sağlamlığını artırır.

4. Metin Tanıma Modeli : PP-OCRv5 tanıma modeli, PP-HGNetV2'yi omurga olarak kullanan çift dallı bir mimariye sahiptir: bir dal (GTC-NRTR) dizi modellemesini geliştirmek için dikkat tabanlı eğitim kullanırken (Hu vd., 2020), diğer dal (SVTR-HGNet) CTC kaybı ile verimli çıkarım üzerine odaklanır. Eğitim sırasında GTC-NRTR dalı SVTR-HGNet dalına rehberlik eder (Du et al. , 2022); ancak tahmin için sadece hafif SVTR-HGNet dalı kullanılarak hem doğruluk hem de hız sağlanır (Li et al., 2022c). Veri oluşturma amacıyla PP-OCRv5, nadir karakterlerin sentez yoluyla üretilenleri de dahil olmak üzere yüksek kaliteli el yazısı örneklerini otomatik olarak etiketlemek ve filtrelemek için geleneksel modelleri ERNIE-4.5-VL-424B-A47B ile birleştirir. Ek olarak, otomatik ayırtırma ve düzenleme mesafesi filtreleme kullanılarak PDF'ler ve e-kitaplar gibi belgelerden büyük ölçekli etiketli veriler elde edilir. Bu veri oluşturma stratejileri, PP-OCRv5'in genel performans iyileştirmesi için sağlam bir veri temeli de sunar.

Buna göre, PP-OCRv5'in temel katkıları şunlardır:

1. Birleşik Çok Dilli Modelleme : PP-OCRv5, Basitleştirilmiş Çince, Geleneksel Çince, Çince Pinyin, İngilizce ve Japonca'nın tek bir model içinde birleşik tanınmasını sağlar. Yenilikçi birleşik mimari tasarımları sayesinde model, 100 MB altında kompakt bir boyutunu korumaktadır . Bu durum, çok dilli senaryolarda birden fazla modelin entegrasyonundan kaynaklanan verimlilik darboğazlarını çözerek endüstriyel dağıtım süreçlerini önemli ölçüde basitleştirmektedir. Bir örnek

¹https://github.com/PaddlePaddle/PaddleClas/blob/release/2.6/docs/en/models/PP-HGNetV2_en.md

illustrating PP-OCRv5's multilingual recognition performance is shown in Figure 4.

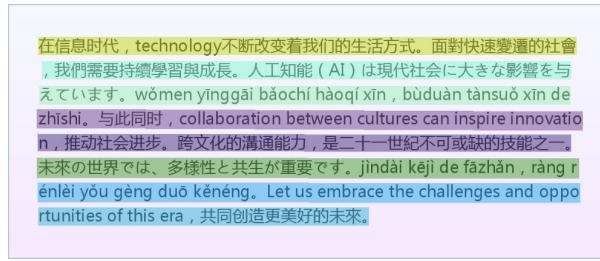


Figure 4 | Multilingual recognition example.

2. Robust Recognition of Complex Handwriting: To address the demands of key application domains such as examination grading in education, bill recognition in finance, and contract entry in the legal sector, PP-OCRv5 has significantly enhanced its capability in handwritten text recognition. Experimental results demonstrate that, compared to previous models, PP-OCRv5 reduces the recognition error rate by 26% on tasks involving non-standard handwriting forms, including both Chinese and English handwritten texts. Figure 5 illustrates the performance of PP-OCRv5 in handwritten text recognition.



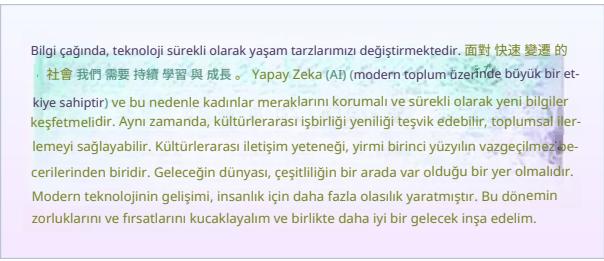
Figure 5 | Handwritten Chinese characters (left) and handwritten English text (right).

3. Robust Recognition of Historical Texts and Uncommon Characters: In complex and non-standard writing scenarios such as Chinese ancient texts and rare Chinese characters, PP-OCRv5 significantly improves text recognition accuracy through optimization of network architecture and systematic construction of diverse, high-quality datasets, effectively meeting the high-precision text recognition requirements across various domains. The recognition performance in complex scenarios is shown in Figure 6.

We evaluated the OCR capabilities across 17 different scenarios, including handwritten Chinese, handwritten English, printed Chinese, printed English, Chinese Pinyin, Japanese, Chinese ancient texts, traditional Chinese, common, blurred, rotated, Greek characters, emojis, tables, artistic fonts, special symbols, and deformed scenes. These diverse scenarios allow us to comprehensively assess the performance and adaptability of the system. Based on the OmniDocBench OCR text evaluation standards, we conducted extensive testing on mainstream OCR methods and multimodal large models. Figure 7 presents the evaluation results, using 1-edit distance as the metric, and lists the average metrics across all scenarios as well as specific performance metric for key scenarios, including handwritten and printed Chinese and English, Chinese Pinyin, and Chinese ancient texts.

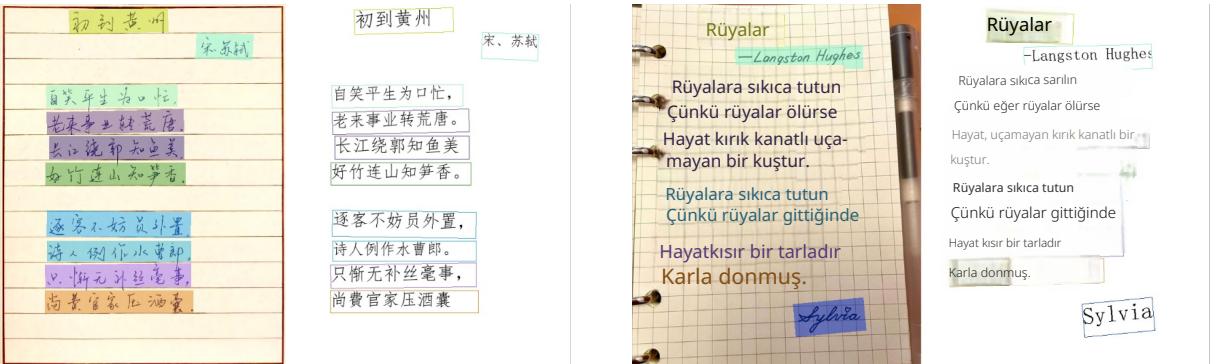
The results indicate that the lightweight PP-OCRv5 ranks first in terms of the average 1-edit

PP-OCRv5'in çok dilli tanıma performansını gösteren örnek Şekil 4'te sunulmuştur.



Şekil 4 | Çok dilli tanıma örneği.

2. Karmaşık El Yazısının Sağlam Tanınması : Eğitimde sınav notlandırma, finansta fatura tanıma ve hukuk sektöründe sözleşme girişi gibi temel uygulama alanlarının taleplerini karşılamak amacıyla PP-OCRv5, el yazısı metin tanıma yeteneğini önemli ölçüde geliştirmiştir. Deneysel sonuçlar, önceki modellere kıyasla PP-OCRv5'in, hem Çince hem de İngilizce el yazısı metinlerini içeren standart dışı el yazısı biçimlerine sahip görevlerde tanıma hata oranını %26 oranında azalttığını göstermektedir. Şekil 5, PP-OCRv5'in el yazısı tanıma performansını göstermektedir.



Şekil 5 | El yazısı Çince karakterler (sol) ve el yazısı İngilizce metin (sağ).

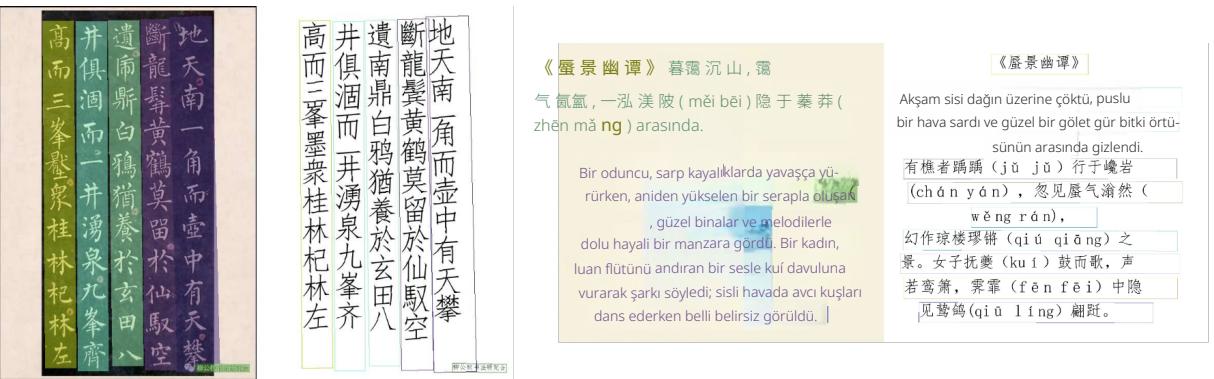
3. Tarihi Metinlerin ve Nadir Karakterlerin Sağlam Tanınması : Çince antik metinler ve nadir Çince karakterler gibi karmaşık ve standart dışı yazım senaryolarında, PP-OCRv5, ağı mimarisinin optimizasyonu ve çeşitli, yüksek kaliteli veri setlerinin sistematik olarak oluşturulması yoluyla metin tanıma doğruluğunu önemli ölçüde artırarak, çeşitli alanlardaki yüksek hassasiyetli metin tanıma gereksinimlerini etkin bir şekilde karşılamaktadır. Karmaşık senaryoların tanıma performansı Şekil 6'da gösterilmiştir.

El yazısı Çince, el yazısı İngilizce, basılı Çince, basılı İngilizce, Çince Pinyin, Japonca, Çince antik metinler, geleneksel Çince, yaygın, bulanık, döndürülmüş, Yunanca karakterler, emojiler, tablolar, sanatsal yazı tipleri, özel semboller ve deform olmuş sahneler dahil olmak üzere 17 farklı senaryoda OCR yeteneklerini değerlendirdik. Bu çeşitli senaryolar, sistemin performansını ve uyarlanabilirliğini kapsamlı bir şekilde değerlendirmemizi sağlamaktadır. OmniDocBench OCR metin değerlendirme standartlarına dayanarak, ana akım OCR yöntemleri ve çok modlu büyük modeller üzerinde kapsamlı testler gerçekleştirdik. Şekil 7, 1-düzenleme mesafesini metrik olarak kullanarak değerlendirme sonuçlarını sunmakta; tüm senaryolardaki ortalama metriklerin yanı sıra el yazısı ve basılı Çince ve İngilizce, Çince Pinyin ve Çince antik metinler dahil olmak üzere anahtar senaryolar için spesifik performans metriklerini listelemektedir.

Sonuçlar, hafif PP-OCRv5'in ortalama 1-düzenleme açısından birinci sırada yer aldığı göstermektedir.



Figure 6 | Vertical Chinese ancient book text (left) and rare Chinese character (right).



Şekil 6 | Dikey Çince eski kitap metni (sol) ve nadir Çince karakter (sağ).

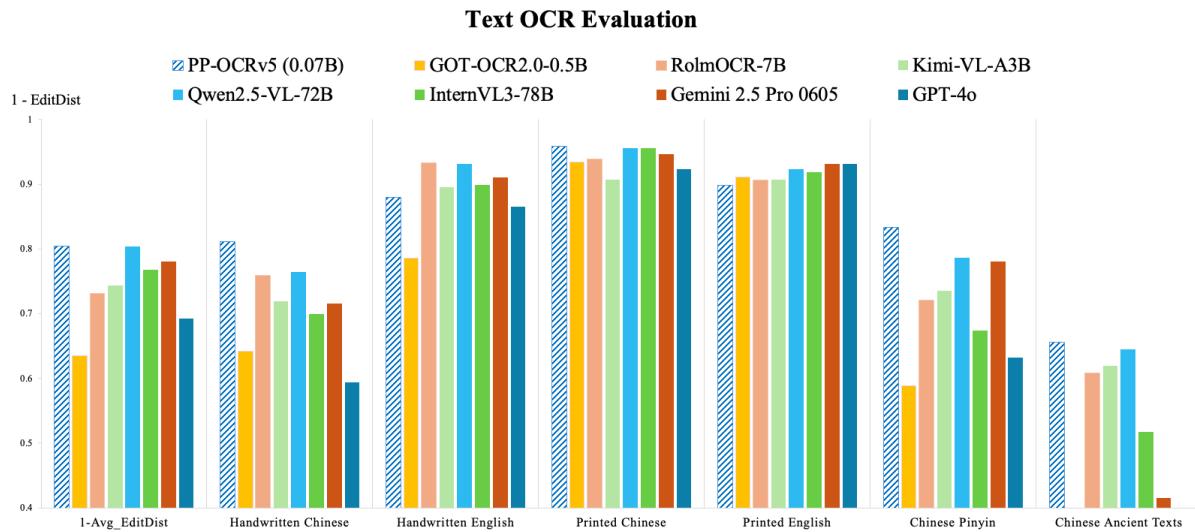
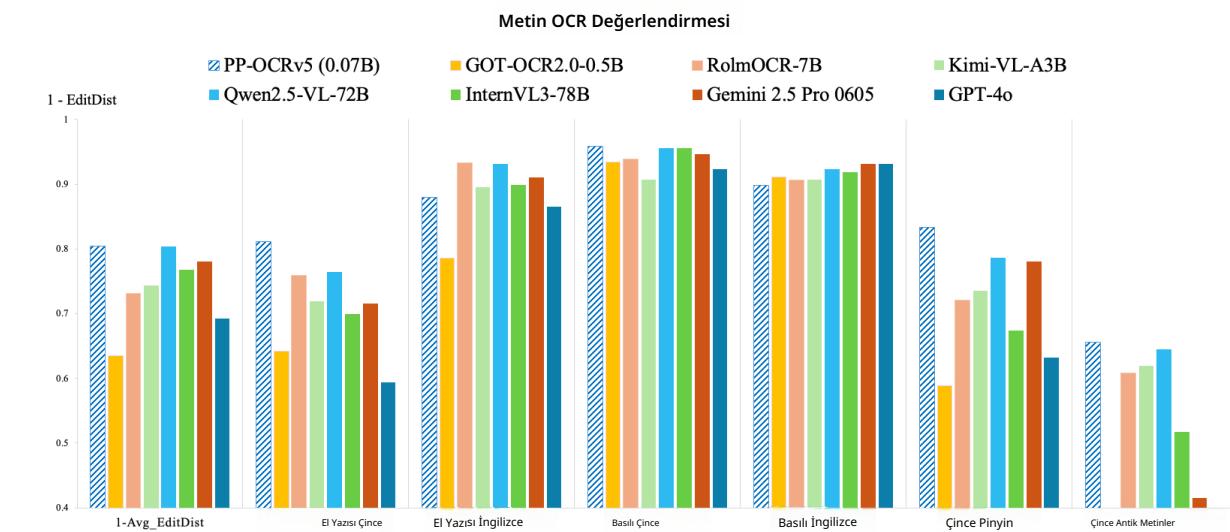


Figure 7 | Text OCR evaluation. The term "1>EditDist" refers to 1 – Edit Distance, a higher value of this metric indicates superior performance.

distance across all scenarios, surpassing all multimodal large models such as GOT-OCR2.0-0.5B (Wei et al., 2024), RolmOCR-7B (AI, 2025), Qwen2.5-VL-72B (Yang et al., 2024), InternVL3-78B (Chen et al., 2024), Gemini 2.5 pro 0605², and GPT-4o³. In Chinese scenarios, whether handwritten or printed, PP-OCRv5 significantly outperforms other methods. Although slightly inferior to multimodal solutions in handwritten English recognition, it is noteworthy that our model has only 0.07 B parameters. In the Chinese Pinyin and Chinese ancient text scenarios, PP-OCRv5 has achieved significant advantages. These results demonstrate that lightweight models specifically designed for OCR tasks can match or even exceed the accuracy of large-scale multimodal models. Simultaneously, they offer substantially reduced computational and storage requirements, significantly enhancing inference efficiency and deployment flexibility, and thereby delivering critical advantages for industrial applications and mobile device deployment.

²<https://deepmind.google/models/gemini/pro/>

³<https://openai.com/index/hello-gpt-4o/>



Şekil 7 | Metin OCR değerlendirme. "1>EditDist" terimi, 1 – Edit Distance'i ifade eder; bu metrikte daha yüksek bir değer, üstün performans anlamına gelir.

tüm senaryolarda mesafe, GOT-OCR2.0-0.5B (Wei ve ark., 2024), RolmOCR-7B (AI, 2025), Qwen2.5-VL-72B (Yang ve ark., 2024), InternVL3-78B (Chen ve ark., 2024), Gemini 2.5 pro 0605² ve GPT-4o³ gibi tüm çok modlu büyük modelleri geride bırakmıştır. Çin senaryolarında, el yazısı veya basılı fark etmeksizin, PP-OCRv5 diğer yöntemlerden ölçüde daha iyi performans göstermektedir. El yazısı İngilizce tanımada çok modlu çözümlerden biraz daha düşük olsa da, modelimizin yalnızca 0.07 B parametreye sahip olması dikkat çekicidir. Çince Pinyin ve Çince antik metin senaryolarında PP-OCRv5, önemli avantajlar elde etmiştir. Bu sonuçlar, OCR görevleri için özel olarak tasarlanmış hafif modellerin, büyük ölçekli çok modlu modellerin doğruluğuna eşit veya hatta daha iyi olabileceğini göstermektedir. Aynı zamanda, ölçüde azaltılmış hesaplama ve depolama gereksinimleri sunarak karışım verimliliğini ve dağıtım esnekliğini büyük ölçüde artırmakta ve böylece endüstriyel uygulamalar ile mobil cihaz dağıtımını için kritik avantajlar sağlama-

²<https://deepmind.google/models/gemini/pro/>

³<https://openai.com/index/hello-gpt-4o/>

2.2. PP-StructureV3

PP-StructureV3 is a multi-model pipeline system developed for document image parsing tasks, it can accurately and efficiently convert document images or PDF files into structured JSON files and Markdown files. As illustrated in the algorithm framework in Figure 8, the system primarily consists of five modules: preprocessing, OCR, layout analysis, document item recognition, and postprocessing.

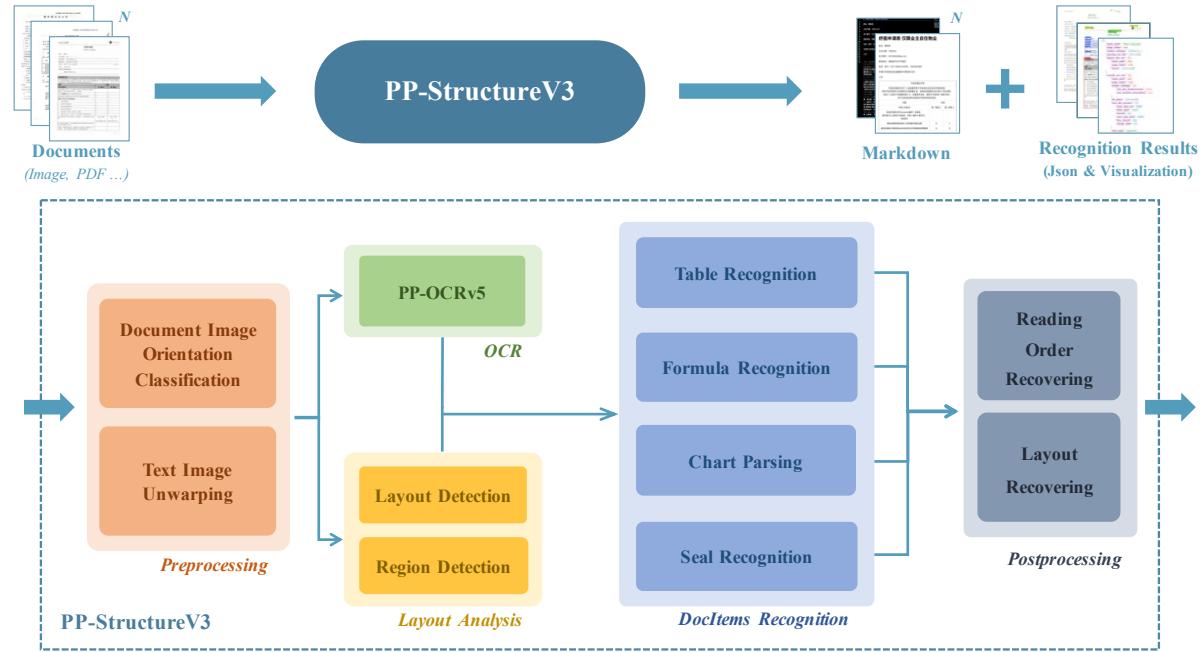


Figure 8 | Pipeline of PP-StructureV3. The pipeline includes Preprocessing, OCRv5, Layout Analysis and Document Items Recognition and Postprocessing. It effectively parses content from images and outputs it as structured data.

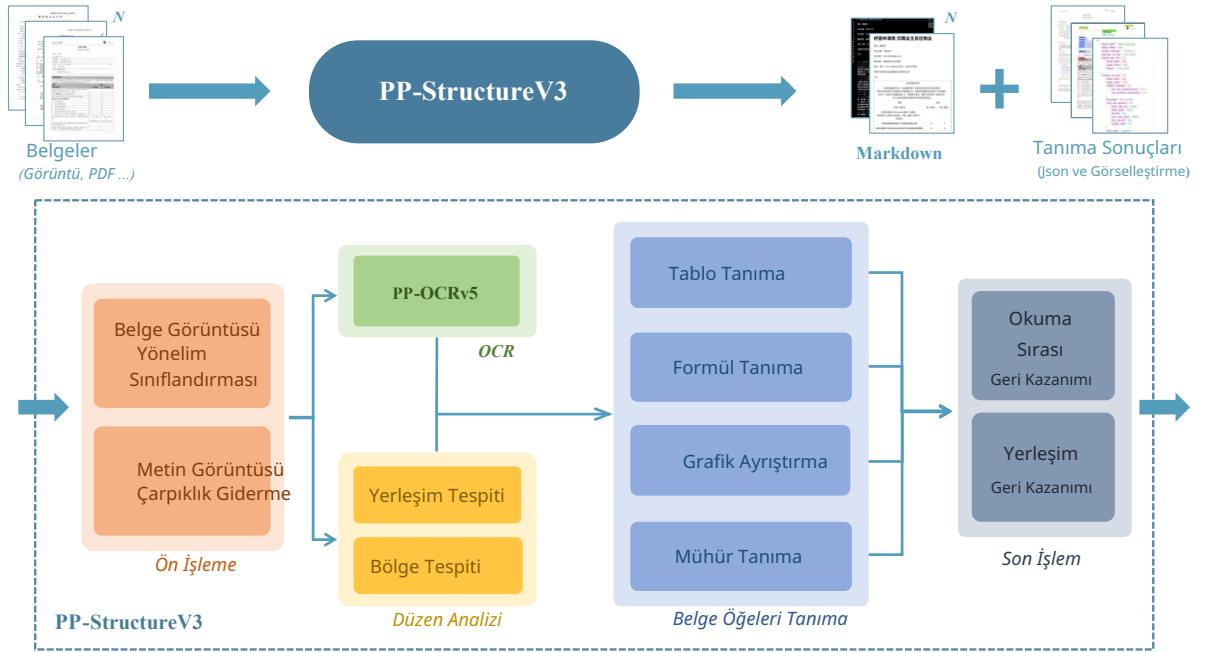
1. Preprocessing: Similar to PP-OCRv5 (see Section 2.1), this module comprises a document image orientation classification model based on PP-LCNet and a text image unwarping model based on UVDoc. It is primarily designed to address issues related to low-quality document images, such as rotation and distortion.

2. OCR: This module employs PP-OCRv5 (see Section 2.1) with preprocessing disabled to detect and recognize all textual content within document images. Compared to PP-OCRv4, PP-OCRv5 achieves significant performance improvements via optimizations in network architecture, training strategies (such as knowledge distillation), and enhancements to the training dataset. Notably, it demonstrates substantial improvements in detection and recognition for challenging scenarios, including vertical text layouts, handwritten text, and rare Chinese characters.

3. Layout Analysis: This module incorporates two models: a layout detection model and a region detection model. The layout detection model, PP-DocLayout-plus, is an optimized version of the PP-DocLayout(Sun et al., 2025). It significantly enhances layout detection performance for complex documents, such as multi-column magazines and newspapers, reports with multiple tables, exams, handwritten documents, Japanese and vertically oriented layouts documents. The newly proposed layout region detection model addresses the problem of multiple articles appearing on a single layout page. For example, a single newspaper page often contains several

2.2. PP-StructureV3

PP-StructureV3, belge görüntüsü ayırtırma görevleri için geliştirilmiş çok modelli bir boru hattı sistemidir, belge görüntülerini veya PDF dosyalarını doğru ve verimli bir şekilde yapılandırılmış JSON dosyalarına ve Markdown dosyalarına dönüştürebilir. Şekil 8'deki algoritma çerçevesinde gösterildiği gibi, sistem temel olarak beş modülden oluşmaktadır: ön işleme, OCR, düzen analizi, belge ögesi tanıma ve son işleme.



Şekil 8 | PP-StructureV3'ün İşlem Hattı. İşlem hattı, Ön İşleme, OCRv5, Yerleşim Analizi, Belge Öğeleri Tanıma ve Son İşlemi içermektedir. İçeriği etkili bir şekilde ayırtırır ve görüntülerden alır ve yapılandırılmış veri olarak çıktısını verir.

1. Ön İşleme : PP-OCRv5'e benzer şekilde (bkz. Bölüm 2.1), bu modül, PP-LCNet tabanlı bir belge görüntüsü yönelik sınıflandırma modeli ve UVDoc tabanlı bir metin görüntüsü bozulma düzeltme modeli içerir. Esas olarak döndürme ve bozulma gibi düşük kaliteli belge görüntüleriyle ilgili sorunları ele almak için tasarlanmıştır.

2. OCR : Bu modül, belge görüntülerindeki tüm metinsel içeriği tespit etmek ve tanımak için ön işleme devre dışı bırakılmış PP-OCRv5'i (bkz. Bölüm 2.1) kullanır. PP-OCRv4 ile karşılaşıldığında, PP-OCRv5, ağ mimarisi, eğitim stratejileri (bilgi damıtma gibi) ve eğitim veri setindeki geliştirmeler aracılığıyla önemli performans iyileştirmeleri sağlamıştır. Özellikle, dikey metin düzenleri, el yazısı metinler ve nadir Çince karakterler gibi zorlu senaryolarda algılama ve tanıma performansında önemli iyileşmeler sergilemektedir.

3. Düzen Analizi : Bu modül, bir düzen algılama modeli ve bir bölge algılama modeli olmak üzere iki modelden oluşmaktadır. Düzen algılama modeli PP-DocLayout-plus, PP-DocLayout'un (Sun ve diğerleri, 2025) optimize edilmiş bir versiyonudur. Çok sütunlu dergiler ve gazeteler, çok sayıda tablo içeren raporlar, sınavlar, el yazısı belgeler, Japonca ve dikey yönelik düzenlere sahip belgeler gibi karmaşık dokümanlar için düzen algılama performansını önemli ölçüde artırmaktadır. Yeni önerilen düzen bölgesi algılama modeli, tek bir düzen sayfasında birden fazla makalenin görünmesi sorununa çözüm sunmaktadır. Örneğin, tek bir gazete sayfası genellikle birkaç

Method Type	Methods	Edit ↓	
		EN	ZH
Pipeline Tools	PP-StructureV3	0.145	0.206
	MinerU-1.3.11 (Wang et al., 2024)	0.166	0.310
	MinerU-0.9.3 (Wang et al., 2024)	0.150	0.357
	Mathpix ¹	0.191	0.365
	Pix2Text-1.1.2.3 (breezedeus, 2022)	0.320	0.528
	Marker-1.2.3 (Paruchuri, 2023)	0.336	0.556
	Unstructured-0.17.2 (Unstructured-IO, 2022)	0.586	0.716
	OpenParse-0.7.0 (Filimoa, 2024)	0.646	0.814
	Docling-2.14.0 (Docling Team, 2024)	0.589	0.909
Expert VLMs	GOT-OCR2.0 (Wei et al., 2024)	0.287	0.411
	Mistral OCR ²	0.268	0.439
	OLMOCR-sqlang (Poznanski et al., 2025)	0.326	0.469
	SmolDocling-256M_transformer (Nassar et al., 2025)	0.493	0.816
	Nougat (Blecher et al., 2023)	0.452	0.973
General VLMs	Gemini2.5-Pro ³	0.148	0.212
	Gemini2.0-flash ⁴	0.191	0.264
	Qwen2.5-VL-72B (Yang et al., 2024)	0.214	0.261
	GPT-4o ⁵	0.233	0.399
	InternVL2-76B (Chen et al., 2024)	0.440	0.443

¹ <https://mathpix.com/>

² <https://mistral.ai/>

³ <https://deepmind.google/models/gemini/pro/>

⁴ <https://deepmind.google/models/gemini/flash/>

⁵ <https://openai.com/index/hello-gpt-4o/>

Table 1 | Comprehensive evaluation of document parsing methods on OmniDocBench.

distinct articles. Using only the layout region detection model makes it challenging to correctly associate elements with their respective articles, leading to errors in reading order recovery. By introducing the layout region detection model, elements can be accurately assigned to their corresponding articles.

4. Document Items Recognition: Based on predictions from the layout detection models, the content of each page element is recognized using appropriate methods, including the table recognition solution PP-TableMagic, the formula recognition model PP-FormulaNet_plus, the chart parsing model PP-Chart2Table, and seal recognition with PP-OCRV4_seal.

- **Table Recognition:** PP-TableMagic is a comprehensive table recognition system composed of several specialized models. It includes a table orientation classification model and a frame type classification model, which determine the rotation and framing style of the table, respectively, guiding the selection of the appropriate recognition method. The cell detection model, based on object detection algorithms, accurately locates individual table cells. Additionally, the structure recognition model outputs the table's structure in HTML format.
- **Formula Recognition:** This model is an enhanced version of PP-FormulaNet ([Liu et al., 2025](#)), capable of recognizing images containing formulas cropped from full document

Yöntem Türü	Yöntemler	Düzenle ↓	
		EN	ZH
Boru Hattı Araçları	PP-StructureV3	0.145	0.206
	MinerU-1.3.11 (Wang ve diğerleri, 2024)	0.166	0.310
	MinerU-0.9.3 (Wang ve diğerleri, 2024)	0.150	0.357
	Mathpix ¹	0.191	0.365
	Pix2Text-1.1.2.3 (breezedeus, 2022)	0.320	0.528
	Marker-1.2.3 (Paruchuri, 2023)	0.336	0.556
	Unstructured-0.17.2 (Unstructured-IO, 2022)	0.586	0.716
	OpenParse-0.7.0 (Filimoa, 2024)	0.646	0.814
	Docling-2.14.0 (Docling Team, 2024)	0.589	0.909
Uzman VLM'ler	GOT-OCR2.0 (Wei vd., 2024)	0.287	0.411
	Mistral OCR ²	0.268	0.439
	OLMOCR-sqlang (Poznanski vd., 2025)	0.326	0.469
	SmolDocling-256M_transformer (Nassar vd., 2025)	0.493	0.816
	Nougat (Blecher vd., 2023)	0.452	0.973
Genel VLM'ler	Gemini2.5-Pro ³	0.148	0.212
	Gemini2.0-flash ⁴	0.191	0.264
	Qwen2.5-VL-72B (Yang vd., 2024)	0.214	0.261
	GPT-4o ⁵	0.233	0.399
	InternVL2-76B (Chen vd., 2024)	0.440	0.443

¹ <https://mathpix.com/>

² <https://mistral.ai/>

³ <https://deepmind.google/models/gemini/pro/>

⁴ <https://deepmind.google/models/gemini/flash/>

⁵ <https://openai.com/index/hello-gpt-4o/>

Table 1 | OmniDocBench üzerinde belge ayırtırma yöntemlerinin kapsamlı değerlendirmesi.

farklı makaleler. Yalnızca düzen bölgesi algılama modelini kullanmak, öğeleri ilgili makaleleriyle doğru bir şekilde ilişkilendirmeyi zorlaştırmakta ve okuma sırası geri kazanımında hatalara yol açmaktadır. Düzen bölgesi algılama modelini kullanarak, öğeler ilgili makalelerine doğru bir şekilde atanabilir.

4. Belge Öğeleri Tanıma : Düzen algılama modellerinden elde edilen tahminlere dayanarak, her sayfa öğesinin içeriği; tablo tanıma çözümü PP-TableMagic, formül tanıma modeli PP-FormulaNet_plus, çizelge ayırtırma modeli PP-Chart2Table ve PP-OCRV4_seal ile mühür tanıma dahil olmak üzere uygun yöntemler kullanılarak tanınır.

- **Tablo Tanıma :** PP-TableMagic, çeşitli uzmanlaşmış modellerden oluşan kapsamlı bir tablo tanıma sistemiidir. Tablonun dönüşünü ve çerçevelene stilini sırasıyla belirleyen bir tablo yönlendirme sınıflandırma modeli ve bir çerçeve tipi sınıflandırma modeli içerir; bu da uygun tanıma yönteminin seçimine rehberlik eder. Nesne algılama algoritmalarına dayanan hücre algılama modeli, tek tek tablo hücrelerini doğru bir şekilde konumlandırır. Ek olarak, yapı tanıma modeli tablonun yapısını HTML formatında çıktı olarak verir.
- **Formül Tanıma :** Bu model, PP-FormulaNet'in ([Liu vd., 2025](#)) geliştirilmiş bir versiyonudur ve tam belgeden kırılmış formül içeren görüntüleri tanıyabilir

images and generating the corresponding LaTeX code. To address the recognition of complex multi-line formulas, the token length was increased to 2560, and the training dataset was expanded to include more complex formulas. Additionally, to handle formulas containing Chinese characters, a large volume of relevant data was mined for training.

- **Chart Parsing:** PP-Chart2Table is a lightweight, end-to-end vision-language model designed to accurately extract data from various types of chart images, such as histograms, line charts, and pie charts, and convert the extracted information into tables represented in markdown format. It genuinely understands and retrieves chart data through an innovative Shuffled Chart Data Retrieval task and meticulous token masking. Its performance is further boosted by a sophisticated data synthesis pipeline that generates diverse, high-quality training data using RAG with high-quality seeds and LLM persona design. A two-stage LLM distillation process, leveraging large volumes of unlabeled out-of-distribution (OOD) data, enhances the model’s adaptability and generalization to real-world scenarios.
- **Seal Recognition:** PP-OCRv4_seal is a system specifically tailored for the recognition of oval, round, and other types of seals. It incorporates a curved text detection model, which can accurately detect and rectify bent text, as well as a general-purpose text recognition model.

5. Post-processing Module: Upon completion of the above steps, the positions and corresponding content of each element in the document are obtained. The post-processing module then reconstructs the relationships among elements, such as linking figures and tables with their captions and recovering the correct reading order. PP-StructureV3 improves upon the X-Y Cut (Ha et al., 1995), significantly enhancing the reconstruction of reading order in complex layouts, including magazines, newspapers, and vertically typeset documents.

To evaluate the performance of PP-StructureV3, we conducted experiments using the OmniDocBench benchmark, with the results presented in Table 1. As shown in the table, PP-StructureV3 demonstrates exceptional performance in Chinese and English document parsing, establishing itself as the current SOTA. It not only significantly outperforms other pipeline-based tools, but also shows strong competitiveness when compared to the most popular expert VLMs and general VLMs.

2.3. PP-ChatOCRv4

PP-ChatOCRv4 is an advanced key information extraction solution for document images, leveraging LLMs, VLMs, and OCR technologies to enable robust key information extraction in challenging scenarios such as complex layouts, multi-page PDFs, rare characters, intricate table structures, and documents containing seals. As illustrated in Figure 9, the overall workflow of PP-ChatOCRv4 comprises the following components: the layout analysis module PP-Structure (Li et al., 2022a), the vector retrieval module, the large language model ERNIE 4.5, the document-oriented vision-language model PP-DocBee2 (Ni et al., 2025), and the result fusion module.

1. **PP-Structure:** PP-Structure serves as the document image parsing module. It is built upon multiple specialized models, including layout detection, text line detection, text recognition, and table structure recognition models. By leveraging these models, PP-Structure is able to parse various elements from document images, thereby generating a structured, text-based representation of the document content.

2. **Vector Retrieval Module:** A feature vector database is constructed using the textual content parsed from document images. During key information extraction, RAG technology

görüntülerini tanıma ve bunlara karşılık gelen LaTeX kodunu üretme yeteneğine sahiptir. Karmaşık çok satırlı formüllerin tanınmasını ele almak amacıyla, belirteç uzunluğu 2560'a çıkarılmış ve eğitim veri seti daha karmaşık formüller içerecek şekilde genişletilmiştir. Ayrıca, Çince karakterler içeren formüller işlemek için büyük hacimli ilgili veriler eğitim amacıyla çıkarılmıştır.

- **Grafik Ayırıştırma :** PP-Chart2Table, histogramlar, çizgi grafikler ve pasta grafikler gibi çeşitli grafik görüntüsü türlerinden verileri doğru bir şekilde çıkarmak ve çıkarılan bilgileri markdown formatında temsil edilen tablolara dönüştürmek için tasarlanmış hafif, uçtan uca bir görme-dil modelidir. Yenilikçi bir Karıştırılmış Grafik Veri Alma görevi ve titiz bir belirteç maskeleme aracılığıyla grafik verilerini gerçekten anlar ve alır. Performansı, RAG ile yüksek kaliteli tohumlar ve LLM kişilik tasarımları kullanarak çeşitli, yüksek kaliteli eğitim verileri üreten gelişmiş bir veri sentezi hattı ile daha da artırılmaktadır. Büyük hacimli etiketlenmemiş dağıtım dışı (OOD) verilerden yararlanan iki aşamalı bir LLM damıtma süreci, modelin gerçek dünya senaryolarına uyaranabilirliğini ve genelleme yeteneğini geliştirir.
- **Mühür Tanıma :** PP-OCRv4_seal, oval, yuvarlak ve diğer mühür türlerinin tanınması için özel olarak tasarlanmış bir sistemdir. Eğri metinleri doğru bir şekilde algılayıp düzeltebilecek kavisli bir metin algılama modeli ve genel amaçlı bir metin tanıma modeli içerir.

5. Son İşlem Modülü : Yukarıdaki adımların tamamlanmasının ardından, belgedeki her bir ögenin konumları ve ilgili içerikleri elde edilir. Son işlem modülü daha sonra, şekilleri ve tabloları alt yazılarıyla ilişkilendirme ve doğru okuma sırasını geri kazanma gibi öğeler arasındaki ilişkileri yeniden yapılandırır. PP-StructureV3, X-Y Kesimi'ni (Ha ve diğerleri, 1995) gerçekleştirerek, dergiler, gazeteler ve dikey olarak dizilmiş belgeler de dahil olmak üzere karmaşık düzenlerde okuma sırasının yeniden yapılandırılmasını önemli ölçüde iyileştirir.

PP-StructureV3'ün performansını değerlendirmek amacıyla, OmniDocBench kıyaslama testini kullanarak deneyler gerçekleştirdik ve sonuçlar Tablo 1'de sunulmuştur. Tabloda görüldüğü üzere, PP-StructureV3, Çince ve İngilizce belge ayırtırmada üstün bir performans sergileyerek mevcut SOTA konumunu almıştır. Diğer ardışık düzen tabanlı araçları önemli ölçüde geride bırakmasının yanı sıra, en popüler uzman VLM'ler ve genel VLM'lerle karşılaşıldığında da güçlü bir rekabet gücü göstermektedir.

2.3. PP-ChatOCRv4

PP-ChatOCRv4, karmaşık düzenler, çok sayfalı PDF'ler, nadir karakterler, karmaşık tablo yapıları ve mühür içeren belgeler gibi zorlu senaryolarda güvenilir anahtar bilgi çıkarımı sağlamak üzere LLM'leri, VLM'leri ve OCR teknolojilerini kullanan, belge görüntüleri için geliştirilmiş bir anahtar bilgi çözümüdür. Şekil 9'da gösterildiği gibi, PP-ChatOCRv4 'ün genel iş akışı şu bileşenlerden oluşmaktadır: düzen analizi modülü PP-Structure (Li et al., 2022a), vektör geri alım modülü, büyük dil modeli ERNIE 4.5, belge odaklı görsel-dil modeli PP-DocBee2 (Ni et al., 2025) ve sonuç birleştirme modülü.

1. **PP-Structure :** PP-Structure, belge görüntüsü ayırtırmaya modülü işlevini görür. Düzen tespiti, metin satırı tespiti, metin tanıma ve tablo yapısı tanıma modelleri dahil olmak üzere birden fazla özel model üzerine inşa edilmiştir. Bu modellerden yararlanarak, PP-Structure belge görüntülerinden çeşitli öğeleri ayırtılabilir, böylece belge içeriğinin yapılandırılmış, metin tabanlı bir temsilini oluşturur.

2. **Vektör Geri Çağırma Modülü :** Belge görüntülerinden ayırtırlan metinsel içerik kullanılarak bir özellik vektör veritabanı oluşturulur. Anahtar bilgi çıkarımı sırasında, RAG teknolojisi

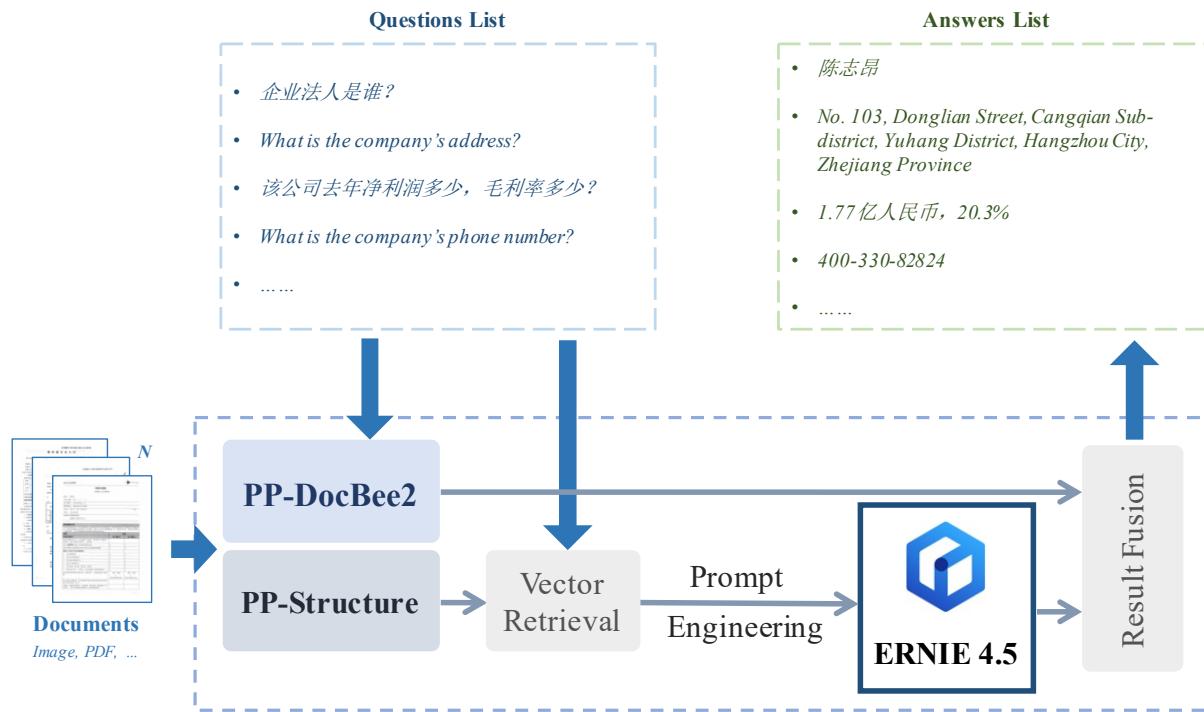


Figure 9 | Pipeline of PP-ChatOCRv4.

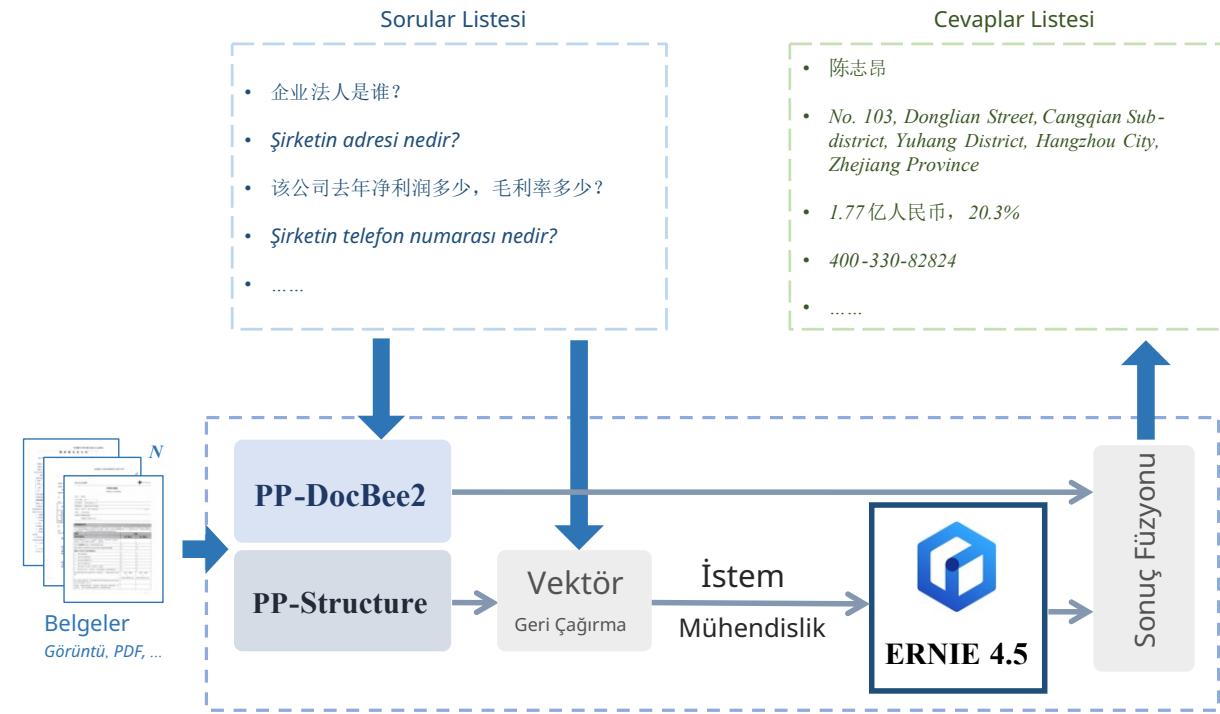
is initially employed to efficiently identify and extract critical information from lengthy and redundant texts. By leveraging RAG, the efficiency and accuracy of information retrieval are significantly enhanced.

3. Large Language Model: The framework supports any large language model (LLM); for example, we currently use ERNIE-4.5-300B-A47B, the latest LLM released by Baidu, to extract information based on carefully designed prompts. These prompts are crafted to seamlessly integrate retrieved textual information with user queries, thereby enhancing both the efficiency and accuracy of the results.

4. PP-DocBee2: The PP-DocBee2 is a novel multimodal large language model with 3 billion parameters designed for end-to-end document image understanding developed by us. It is capable of directly extracting text-based answers from document images using prompts constructed from the given questions.

5. Result Fusion: Extraction results from both text-based and image-based approaches are fused to produce the final output.

To evaluate the performance of PP-ChatOCRv4 and other methods, we designed an end-to-end evaluation pipeline using a custom multi-scenario benchmark dataset comprising 638 document images. These images span a wide range of scenarios, including financial reports, research papers, contracts, manuals, regulations, as well as humanities and science papers. Each document image is accompanied by several questions and corresponding answers, resulting in a total of 1,196 question-answer pairs. The results are presented in Table 2.



Şekil 9 | PP-ChatOCRv4'ün İş Akışı.

uzun ve gereksiz metinlerden kritik bilgileri verimli bir şekilde tanımlamak ve çıkarmak amacıyla başlangıçta kullanılır. RAG'dan yararlanılarak, bilgi kazanımının verimliliği ve doğruluğu önemli ölçüde artırılır.

3. Büyük Dil Modeli : Çerçeve, herhangi bir büyük dil modelini (LLM) destekler; örneğin, özenle tasarlanmış istemlere dayalı bilgi çıkarmak için şu anda Baidu tarafından yayımlanan en son LLM olan ERNIE-4.5-300B-A47B'yi kullanıyoruz. Bu istemler, geri alınan metinsel bilgiyi kullanıcı sorgularıyla sorunsuz bir şekilde entegre etmek üzere tasarlanmıştır, böylece sonuçların hem verimliliğini hem de doğruluğunu artırır.

4. PP-DocBee2 : PP-DocBee2, tarafımızdan geliştirilen, uçtan uca belge görüntüsü analama için tasarlanmış, 3 milyar parametreli yeni bir çok modlu büyük dil modelidir. Verilen sorulardan oluşturulan istemler kullanılarak belge görüntülerinden doğrudan metin tabanlı yanıtları çıkarabilir.

5. Sonuç Füzyonu : Hem metin tabanlı hem de görüntü tabanlı yaklaşımlardan elde edilen çıkışım sonuçları, nihai çıktıyı üretmek üzere birleştirilir.

PP-ChatOCRv4 ve diğer yöntemlerin performansını değerlendirmek amacıyla, 638 belge görüntüsünden oluşan özel bir çok senaryolu kıyaslama veri kümesi kullanarak uçtan uca bir değerlendirme hattı tasarladık. Bu görüntüler; finansal raporlar, araştırma makaleleri, sözleşmeler, kılavuzlar, düzenlemeler ile bilim ve teknoloji makaleleri dahil olmak üzere geniş bir senaryo yelpazesini kapsamaktadır. Her belge görüntüsüne, toplamda 1.196 soru-cevap çifti oluşturan çeşitli sorular ve ilgili cevaplar eşlik etmektedir. Sonuçlar Tablo 2'de sunulmuştur.

Methods	Recall@1
GPT-4o	63.47%
PP-ChatOCRv3	70.08%
Qwen2.5-VL-72B	80.26%
PP-ChatOCRv4	85.55%

Table 2 | Comprehensive evaluation of various solutions on a custom multi-scenario benchmark: overall recall@1 performance based on ground truth comparison.

3. Codebase Architecture Design

3.1. Overall Architecture

The overall architecture of the PaddleOCR 3.0 codebase is illustrated in Figure 10. At its foundation, PaddleOCR 3.0 is built upon the PaddlePaddle framework (Ma et al., 2019), which incorporates a neural network compiler for performance optimization, supplies a highly extensible intermediate representation (IR), and ensures broad hardware compatibility. Building on this foundation, PaddleOCR 3.0 is structured around two core components: a model training toolkit and an inference library. The inference library offers flexible integration paths that naturally extend to deployment. The deployment capabilities will be introduced in Section 4.

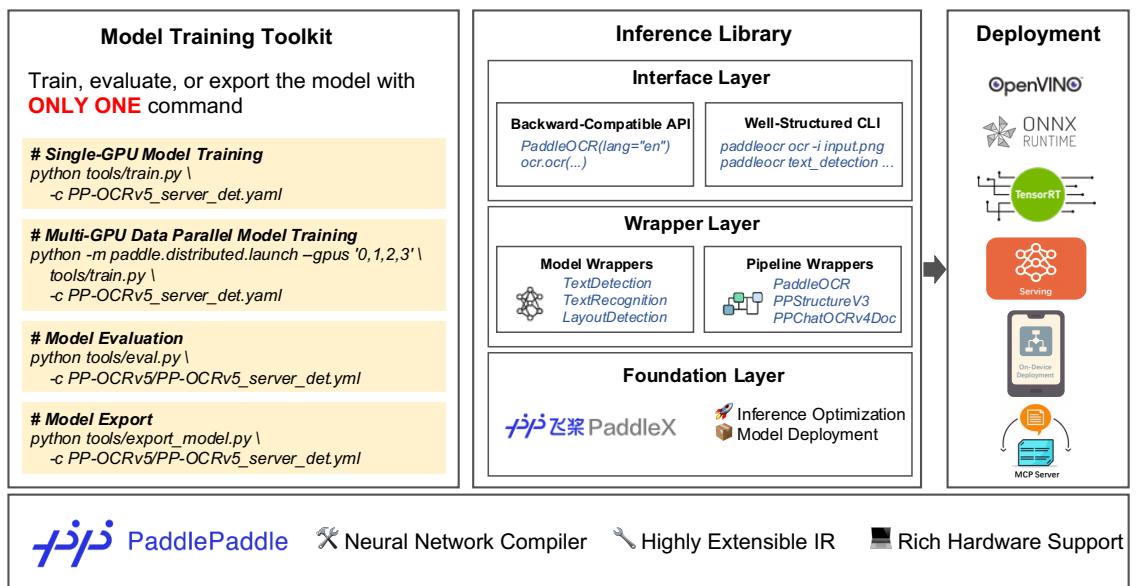


Figure 10 | Overall Architecture of the PaddleOCR 3.0 Codebase.

The training toolkit provides a comprehensive suite of utilities that support the complete training pipeline for various models, including text detection models, text recognition models, etc. It also facilitates the conversion of trained models from dynamic graph format to static graph format, thereby enhancing their suitability for inference and deployment in production environments. Users can execute Python scripts with a single command to perform tasks such as model training, evaluation, and export. Additionally, various parameters can be configured to meet different requirements, such as specifying the path to a pre-trained model or using a custom dataset directory.

Complementing this, the inference library is designed to be lightweight and highly efficient.

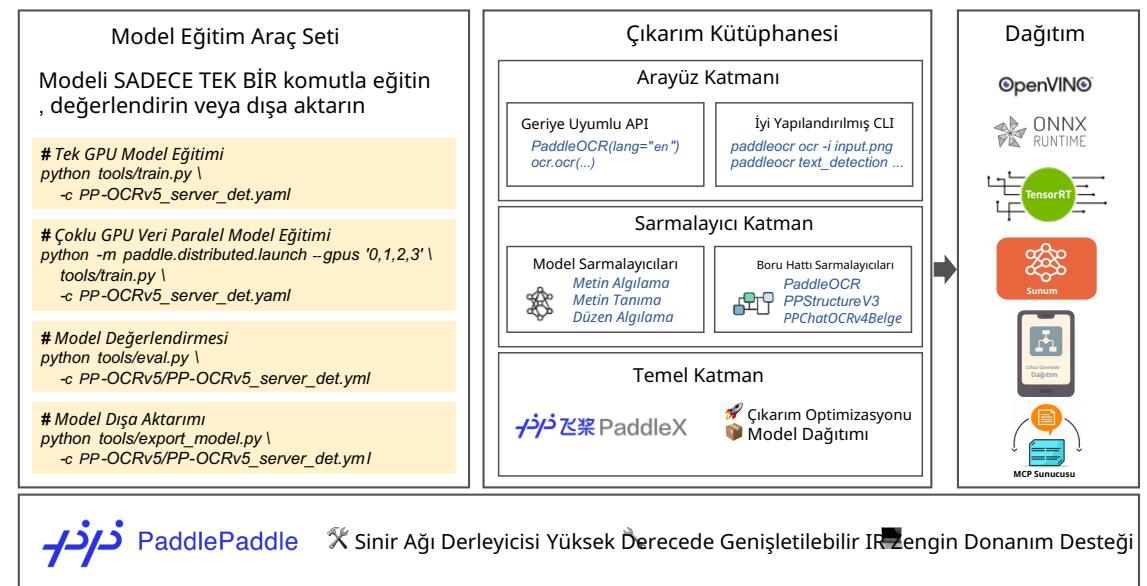
Yöntemler	Recall@1
GPT-4o	63.47%
PP-ChatOCRv3	70.08%
Qwen2.5-VL-72B	80.26%
PP-ChatOCRv4	85.55%

Tablo 2 | Özel bir çok senaryolu kıyaslama üzerinde çeşitli çözümlerin kapsamlı değerlendirmesi: gerçek değer karşılaştırmasına dayalı genel recall@1 performansı.

3. Kod Tabanı Mimari Tasarımı

3.1. Genel Mimari

PaddleOCR 3.0 kod tabanının genel mimarisini Şekil 10'da gösterilmiştir. Temelinde, PaddleOCR 3.0, performans optimizasyonu için bir sinir ağı derleyicisi içeren, yüksek derecede genişletilebilir bir ara temsil (IR) sağlayan ve geniş donanım uyumluluğu garantileyen PaddlePaddle çatısı (Ma vd., 2019) üzerine inşa edilmiştir. Bu temel üzerine inşa edilen PaddleOCR 3.0, bir model eğitim araç seti ve bir çıkışım kütüphanesi olmak üzere iki ana bileşen etrafında yapılandırılmıştır. Çıkışım kütüphanesi, doğal olarak dağıtıma uzanan esnek entegrasyon yolları sunar. Dağıtım yetenekleri Bölüm 4'te tanıtılacaktır.



Şekil 10 | PaddleOCR 3.0 Kod Tabanının Genel Mimarisi.

Eğitim araç seti, metin algılama modelleri ve metin tanıma modelleri gibi çeşitli modeller için eksiksiz eğitim hattını destekleyen kapsamlı bir yardımcı program paketi sunar. Ayrıca, eğitilmiş modellerin dinamik grafik formatından statik grafik formatına dönüştürülmesini kolaylaştırarak, üretim ortamlarında çıkışım ve dağıtım için uygunluklarını artırır. Kullanıcılar, model eğitimi, değerlendirme ve dışa aktarımı gibi görevleri tek bir komutla gerçekleştirmek üzere Python betiklerini çalıştırabilirler. Ek olarak, önceden eğitilmiş bir modelin yolunu belirtmek veya özel bir veri kümlesi dizini kullanmak gibi farklı gerekşimleri karşılamak amacıyla çeşitli parametreler yapılandırılabilir.

Buna ek olarak, çıkışım kütüphanesi hafif ve oldukça verimli olacak şekilde tasarlanmıştır.

It supports loading both officially released inference models and custom models trained by users. The library enables inference across eight end-to-end model pipelines and can be readily integrated into real-world applications. We will elaborate the design of the inference library in the next subsection. The inference library serves as the foundation for downstream deployment capabilities, including high-performance inference across various frameworks, deploying the model pipeline as a service, deploying it to mobile devices, and running it via a Model Context Protocol (MCP) server.

3.2. Inference Library Design

In this subsection, we present the rationale behind the design and structural organization of the PaddleOCR 3.0 inference library.

Let us start with the defects of the inference library in PaddleOCR 2.x:

- All parameters of the PaddleOCR 2.x CLI exist in the same namespace. This is not extensible, as each new feature required manually adding more arguments to an increasingly bloated global parameter list, making it difficult to maintain and scale.
- In PaddleOCR 2.x, the parameters are only configurable through function arguments. Such an approach hinders reproducibility and portability, especially in scenarios where configurations need to be shared or version-controlled.
- PaddleOCR 2.x lacks clear separation of concerns—the boundary between the model development toolkit (primarily designed for training) and the inference library was not clearly defined. Instead, the inference library was built directly on top of the model development toolkit inference scripts, which introduced two entry points for the inference functionality, potentially causing confusion for users. Additionally, the inference library was constructed based on assumptions about what the entry points should be, which broke modularity and maintainability.

To address these issues, we upgraded the inference library on top of the PaddleX 3.0 toolkit⁴, which provides extensive inference optimization and deployment features. The backward compatibility was also considered, which minimizes migration effort for users transitioning from PaddleOCR 2.x. The new inference library consists of three layers, as illustrated in Figure 10.

- **Interface Layer:** The library offers both a Python API and a CLI for user interaction. In PaddleOCR 3.0, all OCR tasks are accessed through a consistent and unified Python API. To facilitate a smooth transition for existing users, the API preserves backward compatibility for key methods and parameters. The CLI has been completely redesigned compared to PaddleOCR 2.x, introducing subcommands that clearly distinguish between different tasks and thereby providing a cleaner, more intuitive user experience.
- **Wrapper Layer:** This layer offers Pythonic wrappers for core PaddleX components, including models and pipelines. These wrappers deliver unified interfaces and flexible configuration management. In addition to maintaining the backward compatibility in the argument-based approach previously preferred in PaddleOCR 2.x, the PaddleX-style configuration file-based approach is also supported, which allows configurations to be stored and reused in a portable and reproducible manner.

⁴<https://github.com/PaddlePaddle/PaddleX/tree/release/3.0>

Hem resmi olarak yayımlanan çıkarım modellerini hem de kullanıcılar tarafından eğitilen özel modelleri yüklemeyi destekler. Kütüphane, sekiz uçtan uca model hattı boyunca çıkarım yapılmasını sağlar ve gerçek dünya uygulamalarına kolayca entegre edilebilir. Çıkarım kütüphanesinin tasarımını bir sonraki alt bölümde detaylandıracız. Çıkarım kütüphanesi, çeşitli çerçevelerde yüksek performanslı çıkarım, model hattının bir hizmet olarak dağıtılması, mobil cihazlara dağıtılması ve bir Model Bağlam Protokolü (MCP) sunucusu aracılığıyla çalıştırılması gibi alt akış dağıtım yetenekleri için temel oluşturur.

3.2. Çıkarım Kütüphanesi Tasarımı

Bu alt bölümde, PaddleOCR 3.0 çıkarım kütüphanesinin tasarım ve yapısal organizasyonununardındaki mantığı sunmaktadır.

PaddleOCR 2.x'teki çıkarım kütüphanesinin eksiklikleriyle başlayalım:

- PaddleOCR 2.x CLI'nin tüm parametreleri aynı ad alanında bulunmaktadır. Her yeni özelliğin, giderek genişleyen genel bir parametre listesine manuel olarak daha fazla argüman eklenmesini gerektirmesi nedeniyle bu durum genişletilebilir değildir, bu da bakımı ve ölcüklenirilmesini zorlaştırır.
- PaddleOCR 2.x'te parametreler yalnızca fonksiyon argümanları aracılığıyla yapılandırılabilirliktedir. Bu tür bir yaklaşım, özellikle konfigürasyonların paylaşılması veya sürüm kontrolü altında tutulması gereken senaryolarda tekrarlanabilirliği ve taşınabilirliğini engellemektedir.
- PaddleOCR 2.x, sorumluluklarının net bir şekilde ayrılmasıından yoksundu; model geliştirme araç seti (öncelikli olarak eğitim için tasarlanmış) ile çıkarım kütüphanesi arasındaki sınır açıkça tanımlanmamıştı. Bunun yerine, çıkarım kütüphanesi doğrudan model geliştirme araç seti çıkarım betikleri üzerine inşa edilmiştir ve bu durum, çıkarım işlevselligi için iki giriş noktası sunarak kullanıcılar için potansiyel olarak kafa karışıklığına yol açmaktadır. Ek olarak, çıkarım kütüphanesi, giriş noktalarının ne olması gerektiğine dair varsayımlar üzerine inşa edilmiştir; bu durum, modülerliği ve sürdürülebilirliği bozdu.

Bu sorunları gidermek amacıyla, çıkarım kütüphanesini, kapsamlı çıkarım optimizasyonu ve dağıtım özellikleri sunan PaddleX 3.0 araç kiti⁴ üzerine yükseltti. PaddleOCR 2.x'ten geçiş yapan kullanıcılar için migrasyon çabasını en aza indiren geriye dönük uyumluluk da göz önünde bulunduruldu. Yeni çıkarım kütüphanesi, Şekil 10'da gösterildiği gibi üç katmandan oluşmaktadır.

- **Arayüz Katmanı :** Kütüphane, kullanıcı etkileşimi için hem bir Python API'si hem de bir CLI sunar. PaddleOCR 3.0'da tüm OCR görevlerine tutarlı ve birleşik bir Python API aracılığıyla erişilir. Mevcut kullanıcılar için sorunsuz bir geçiş kolaylaştırılmıştır amaçyla, API, temel yöntemler ve parametreler için geriye dönük uyumluluğu korur. CLI, PaddleOCR 2.x'e kıyasla tamamen yeniden tasarlandı ve farklı görevler arasında net bir ayrımlı yapan alt komutlar sunarak daha temiz, daha sezgisel bir kullanıcı deneyimi sağladı.
- **Sarmalayıcı Katmanı :** Bu katman, modeller ve işlem hatları dahil olmak üzere temel PaddleX bileşenleri için Pythonik sarmalayıcılar sunar. Bu sarmalayıcılar, birleşik arayüzler ve esnek yapılandırma yönetimi sağlar. Daha önce PaddleOCR 2.x'te tercih edilen argüman tabanlı yaklaşımda geriye dönük uyumluluğun korunmasının yanı sıra, yapılandırmaların taşınabilir ve tekrarlanabilir bir şekilde saklanması ve yeniden kullanılmasını sağlayan PaddleX tarzı yapılandırma dosyası tabanlı yaklaşım da desteklenmektedir.

⁴<https://github.com/PaddlePaddle/PaddleX/tree/release/3.0>

- Foundation Layer:** At the foundation lies the PaddleX 3.0 toolkit, which forms the core of PaddleOCR 3.0. It offers powerful features for inference optimization and model deployment, which are fully integrated into PaddleOCR 3.0. Transferring the basis from PaddleOCR scripts to PaddleX ensures the separation of roles of the model training toolkit and the inference library, eliminating redundant entry points and clarifying functional boundaries. This decoupling allows each component to evolve independently, reduces user confusion, and lays the foundation for a more robust and maintainable system design.

This layered architecture ensures that higher-level components depend only on lower-level abstractions, promoting loose coupling, modularity, and ease of maintenance.

4. Deployment

An overview of the deployment capabilities of PaddleOCR 3.0 is depicted in Figure 11. To support a wide range of application scenarios, PaddleOCR 3.0 offers flexible and comprehensive deployment options, including high-performance inference, serving, and on-device deployment. In real-world production environments, OCR-related systems are often subject to constraints beyond recognition accuracy, such as latency, throughput, and hardware compatibility. PaddleOCR addresses these requirements by providing configurable deployment tools that simplify integration across various platforms. In addition, to facilitate integration with LLM applications, PaddleOCR provides an MCP server, which allows users to leverage high-performance inference pipelines or pipeline servers.

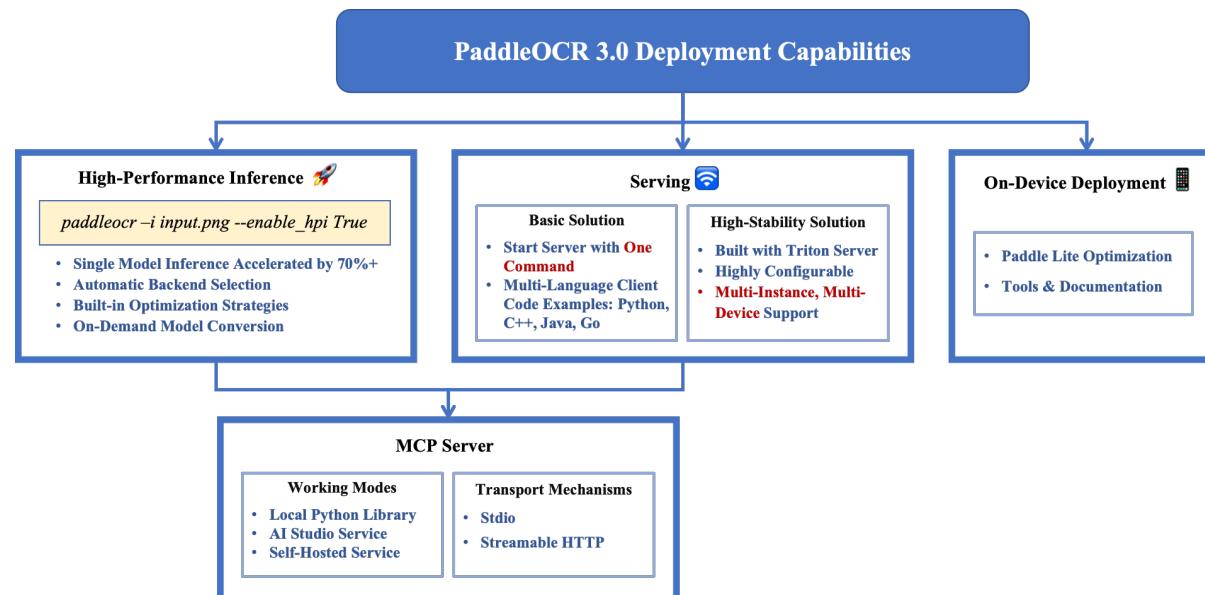


Figure 11 | Overview of the deployment capabilities of PaddleOCR 3.0. PaddleOCR 3.0 provides high-performance inference, serving, and on-device deployment capabilities.

Additionally, it enables users to easily deploy an MCP server based on PaddleOCR.

4.1. High-Performance Inference

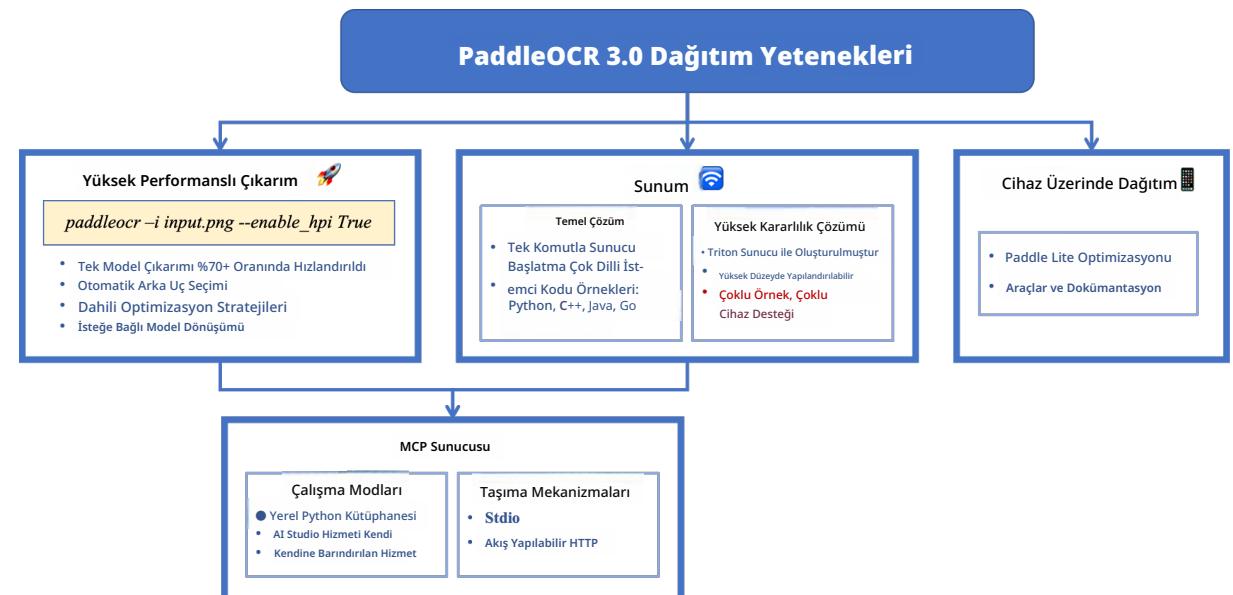
PaddleOCR 3.0 provides the high-performance inference feature that enables users to optimize runtime performance without the need to manually tune low-level configurations. High-performance inference provides notable acceleration for some key models. For instance, on

- Temel Katman :** Temelde PaddleOCR 3.0'ın çekirdeğini oluşturan PaddleX 3.0 araç seti yer almaktadır. Çıkarım optimizasyonu ve model dağıtımını için güçlü özellikler sunar ve bunlar PaddleOCR 3.0'a tam olarak entegre edilmiştir. Temelin PaddleOCR betiklerinden PaddleX'e aktarılması, model eğitim araç setinin ve çıkış kütüphanesinin rollerinin ayırmasını sağlayarak, gereksiz giriş noktalarını ortadan kaldırır ve işlevsel sınırları netleştirir . Bu ayrışma, her bir bileşenin bağımsız olarak gelişmesine olanak tanır, kullanıcı karmaşasını azaltır ve daha sağlam ve sürdürülebilir bir sistem tasarımları için temel oluşturur.

Bu katmanlı mimari, üst düzey bileşenlerin yalnızca alt düzey soyutlamalara bağlı olmasına sağlayarak, gevşek bağlantılı, modülerliği ve bakım kolaylığını teşvik eder.

4. Dağıtım

PaddleOCR 3.0'ın dağıtım yeteneklerine genel bir bakış Şekil 11'de sunulmuştur. Geniş bir uygulama senaryosunu yelpazesini desteklemek amacıyla PaddleOCR 3.0, yüksek performanslı çıkış, sunum ve cihaz içi dağıtım dahil olmak üzere esnek ve kapsamlı dağıtım seçenekleri sunar. Gerçek dünya üretim ortamlarında, OCR ile ilgili sistemler genellikle tanıma doğruluğunu ötesinde gecikme süresi, iş hacmi ve donanım uyumluluğu gibi kısıtlamalara tabidir. PaddleOCR, çeşitli platformlarda entegrasyonu basitleştiren yapılandırılabilir dağıtım araçları sağlayarak bu gereksinimleri karşılar. Ek olarak, LLM uygulamalarıyla entegrasyonu kolaylaştırmak için PaddleOCR, kullanıcıların yüksek performanslı çıkışım işlem hatlarını veya işlem hattı sunucusunu kullanmalarına olanak tanıyan bir MCP sunucusu sağlar.



Şekil 11 | PaddleOCR 3.0'ın dağıtım yeteneklerine genel bakış. PaddleOCR 3.0, yüksek performanslı çıkış, sunum ve cihaz içi dağıtım yetenekleri sunar.

Ayrıca, kullanıcıların PaddleOCR tabanlı bir MCP sunucusunu kolayca dağıtmalarını sağlar.

4.1. Yüksek Performanslı Çıkarım

PaddleOCR 3.0, kullanıcıların düşük seviyeli yapılandırmaları manuel olarak ayarlama ihtiyacı duymadan çalışma zamanı performansını optimize etmelerini sağlayan yüksek performanslı çıkış özelliği sunar. Yüksek performanslı çıkış, bazı temel modeller için kayda değer bir hızlanma sağlar. Örneğin,

NVIDIA Tesla T4 devices, enabling high-performance inference reduces the single-model inference latency of PP-OCRv5_mobile_rec by 73.1% and that of PP-OCRv5_mobile_det by 40.4%. The key features of PaddleOCR 3.0's high-performance inference capability include:

- Automatic selection of appropriate inference backends based on the runtime environment and model characteristics, including support for Paddle Inference, OpenVINO ([Intel Corporation, 2018](#)), ONNX Runtime ([Microsoft Corporation, 2018](#)), and TensorRT ([NVIDIA Corporation, 2017](#)).
- Built-in optimization strategies such as multi-threading and FP16 inference to better utilize hardware resources.
- On-demand model conversion from PaddlePaddle static graphs to ONNX format to enable acceleration on compatible inference engines.

Users can easily achieve inference acceleration by enabling the `enable_hpi` switch, while all underlying optimization details are managed by PaddleOCR. For advanced needs, PaddleOCR also supports fine-grained tuning of high-performance inference configurations in pipelines by passing Python/command line parameters or modifying configuration files.

4.2. Serving

PaddleOCR 3.0 supports pipeline serving for building scalable and production-ready OCR-related services. Two solutions are provided:

- **Basic Serving:** A lightweight solution based on FastAPI ([Ramírez, 2018](#)) with minimal setup, suitable for rapid validation and scenarios with low concurrency requirements. For this solution, users can run any pipeline as a service with a single command via the CLI. For the client-side code, PaddleOCR 3.0 provides rich calling examples in seven programming languages: Python, C++, Java, Go, C#, Node.js, and PHP. Users can refer to these examples to quickly integrate the service capabilities into their own applications.
- **High-Stability Serving:** A more robust option built on NVIDIA Triton Inference Server ([NVIDIA Corporation, 2018](#)), which supports more advanced deployment configurations. This solution is suitable for scenarios with higher requirements for stability and performance. For example, the server can be configured to run multiple instances across multiple GPUs to fully utilize the available computing resources.

Both solutions share similar interfaces. Users can start with the basic solution for rapid validation, and then decide whether to adopt the more complex high-stability solution according to their needs, usually without significant migration costs.

4.3. On-Device Deployment

To support the deployment on resource-constrained devices, PaddleOCR 3.0 enables deployment of PP-OCR models on mobile platforms. It provides supporting tools and documentation for model optimization and integration with the Paddle-Lite⁵, runtime, making it feasible to run OCR tasks efficiently on mobile devices.

⁵<https://github.com/PaddlePaddle/Paddle-Lite>

NVIDIA Tesla T4 cihazları, yüksek performanslı çıkışına olanak tanıarak PP-OCRv5_mobile_rec'in tek model çıkışım gecikmesini %73,1 ve PP-OCRv5_mobile_det'inkini %40,4 oranında azaltır. PaddleOCR 3.0'ın yüksek performanslı çıkışım yeteneğinin temel özellikleri şunlardır:

- Paddle Inference, OpenVINO ([Intel Corporation, 2018](#)), ONNX Runtime ([Microsoft Corporation, 2018](#)) ve TensorRT ([NVIDIA Corporation, 2017](#)) desteği dahil olmak üzere, çalışma zamanı ortamına ve model özelliklerine göre uygun çıkışım arkalarının otomatik seçimi.
- Donanım kaynaklarını daha iyi kullanmak için çoklu iş parçacığı ve FP16 çıkışımı gibi yerleşik optimizasyon stratejileri.
- Uyumlu çıkışım motorlarında hızlandırmayı sağlamak amacıyla PaddlePaddle statik grafiklerinden ONNX formatına istege bağlı model dönüşümü.

Kullanıcılar, `enable_hpi` anahtarını etkinleştirerek çıkışım hızlandırmasını kolayca sağlayabiliyorlarken, tüm temel optimizasyon ayrıntıları PaddleOCR tarafından yönetilir. Gelişmiş ihtiyaçlar için PaddleOCR, Python/komut satırı parametreleri ileteerek veya yapılandırma dosyalarını değiştirerek boru hatlarında yüksek performanslı çıkışım yapılandırmalarının ince ayarını da destekler.

4.2. Sunum

PaddleOCR 3.0, ölçeklenebilir ve üretime hazır OCR ile ilgili hizmetler oluşturmak için boru hattı sunumunu destekler. İki çözüm sunulmaktadır:

- **Temel Sunum :** Hızlı doğrulama ve düşük eşzamanılık gereksinimleri olan senaryolar için uygun, minimum kurulumla FastAPI'ye ([Ramírez, 2018](#)) dayalı hafif bir çözüm. Bu çözüm için kullanıcılar, CLI aracılığıyla tek bir komutla herhangi bir boru hattını hizmet olarak çalıştırabilirler. İstemci tarafı kod için PaddleOCR 3.0, yedi programlama dilinde zengin çağrı örnekleri sunar: Python, C++, Java, Go, C#, Node.js ve PHP. Kullanıcılar, hizmet yeteneklerini kendi uygulamalarına hızlı bir şekilde entegre etmek için bu örneklerle başvurabilirler.
- **Yüksek Kararlılıklı Sunum :** NVIDIA Triton Inference Server ([NVIDIA Corporation, 2018](#)) üzerinde inşa edilmiş, daha gelişmiş dağıtım yapılandırmalarını destekleyen daha sağlam bir seçenek. Bu çözüm, kararlılık ve performans için daha yüksek gereksinimleri olan senaryolar için uygundur. Örneğin, sunucu, mevcut bilgi işlem kaynaklarını tam olarak kullanmak amacıyla birden fazla GPU'da birden fazla örnek çalışacak şekilde yapılandırılabilir.

Her iki çözüm de benzer arayuzleri paylaşmaktadır. Kullanıcılar, hızlı doğrulama için temel çözümle başlayabilir ve ardından genellikle önemli geçiş maliyetleri olmaksızın ihtiyaçlarına göre daha karmaşık yüksek kararlılıklı çözümü benimseyip benimsememeye karar verebilirler.

4.3. Cihaz Üzerinde Dağıtım

Kaynak kısıtlı cihazlarda dağıtımları desteklemek için PaddleOCR 3.0, PP-OCR modellerinin mobil platformlarda dağıtımını sağlar. Model optimizasyonu ve Paddle-Lite⁵ çalışma zamanı ile entegrasyon için destekleyici araçlar ve belgeler sağlayarak, mobil cihazlarda OCR görevlerini verimli bir şekilde çalıştırmayı mümkün kılar.

⁵<https://github.com/PaddlePaddle/Paddle-Lite>

4.4. MCP Server

PaddleOCR 3.0 provides a lightweight MCP server, enabling smooth integration of PaddleOCR's core capabilities into any MCP-compatible host. Both the OCR and PP-StructureV3 pipelines are currently accessible as tools via the MCP server.

Built on top of the PaddleOCR inference library, the MCP server supports various inference and deployment methods provided by PaddleOCR. At present, it can operate in one of three working modes:

- **Local:** Runs the PaddleOCR pipeline directly on the local machine using the installed Python library. This mode is suitable for offline usage and situations with strict data privacy requirements. High-performance inference can be activated to accelerate the inference process.
- **AI Studio:** Utilizes cloud services hosted by the PaddlePaddle AI Studio community ([PaddlePaddle Team, 2019](#)). This mode is ideal for quickly trying out features, validating solutions, and for no-code development scenarios.
- **Self-Hosted:** Connects to a user-hosted PaddleOCR service. This mode offers the advantages of pipeline serving and high flexibility, making it well-suited for scenarios requiring custom service configurations.

Regardless of the selected working mode, setting up the PaddleOCR MCP server is straightforward for users. Example configuration files are included in the appendix 5 for reference. Additionally, the PaddleOCR MCP server supports both stdio and Streamable HTTP transport mechanisms, offering flexibility for a wide range of deployment scenarios. With its adaptable architecture and support for multiple deployment modes, the PaddleOCR MCP server can effectively address diverse real-world application needs, delivering robust and scalable solutions for both individual developers and enterprise users.

5. Conclusion

PaddleOCR has been dedicated to the field of OCR and document parsing for many years, aiming to provide more valuable technical solutions. PaddleOCR 3.0 is a milestone upgrade, with technologies like PP-OCRv5, PP-StructureV3, and PP-ChatOCRv4 set to play a significant role in the era of large-scale models. Moving forward, we will continue to expand our models, including the upcoming release of multilingual text recognition models, multimodal OCR, and document parsing models. If you find PaddleOCR 3.0 useful or wish to use it in your projects, please kindly cite this technical report.

References

- J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 technical report. [arXiv preprint arXiv:2303.08774](#), 2023.
- R. AI. Rolmocr: A faster, lighter open source ocr model, 2025.
- Baidu-ERNIE-Team. Ernie 4.5 technical report, 2025.
- L. Blecher, G. Cucurull, T. Scialom, and R. Stojnic. Nougat: Neural optical understanding for academic documents, 2023.

4.4. MCP Sunucusu

PaddleOCR 3.0, PaddleOCR'ın temel yeteneklerinin herhangi bir MCP uyumlu ana bilgisayara sorunsuz entegrasyonunu sağlayan hafif bir MCP sunucusu sunar. Hem OCR hem de PP-StructureV3 işlem hatları, şu anda MCP sunucusu üzerinden araç olarak erişilebilir durumdadır.

PaddleOCR çıkışım kütüphanesi üzerine inşa edilen MCP sunucusu, PaddleOCR tarafından sağlanan çeşitli çıkışım ve dağıtım yöntemlerini desteklemektedir. Şu anda üç farklı çalışma modunda faaliyet gösterebilir:

- **Yerel:** Kurulu Python kütüphanesini kullanarak PaddleOCR işlem hattını doğrudan yerel makinede çalıştırır. Bu mod, çevrimdışı kullanım ve katı veri gizliliği gereklilikleri olan durumlar için uygundur. Çıkışım sürecini hızlandırmak amacıyla yüksek performanslı çıkışım etkinleştirilebilir.
- **AI Studio:** PaddlePaddle AI Studio topluluğu ([PaddlePaddle Ekibi, 2019](#)) tarafından barındırılan bulut hizmetlerini kullanır. Bu mod, özellikleri hızlıca denemek, çözümleri doğrulamak ve kodlu geliştirme senaryoları için idealdir.
- **Kendi Barındırma:** Kullanıcı tarafından barındırılan bir PaddleOCR hizmetine bağlanır. Bu mod, işlem hattı sunumu ve yüksek esneklik avantajları sunarak, özel hizmet yapılandırmaları gerektiren senaryolar için oldukça uygundur.

Seçilen çalışma modundan bağımsız olarak, PaddleOCR MCP sunucusunun kurulumu kullanıcılar için basittir. Örnek yapılandırma dosyaları, referans amacıyla ek 5'te sunulmuştur. Ek olarak, PaddleOCR MCP sunucusu hem stdio hem de Akışkan HTTP taşıma mekanizmalarını destekleyerek geniş bir dağıtım senaryosu yelpazesi için esneklik sunar. Uyarlanabilir mimarisi ve birden fazla dağıtım modunu desteklemesiyle PaddleOCR MCP sunucusu, çeşitli gerçek dünya uygulama ihtiyaçlarını etkin bir şekilde karşılayabilir; hem bireysel geliştiriciler hem de kurumsal kullanıcılar için sağlam ve ölçeklenebilir çözümler sunar.

5. Sonuç

PaddleOCR, daha değerli teknik çözümler sunmayı hedefleyerek uzun yıllardır OCR ve belge ayrıştırma alanına adanmıştır. PaddleOCR 3.0, büyük ölçekli modeller çağında PP-OCRv5, PP-StructureV3 ve PP-ChatOCRv4 gibi teknolojilerin önemli bir rol oynayacağı dönüm noktası niteliğinde bir yükseltmedir. İlerleyen süreçte, çok dilli metin tanıma modelleri, çok modlu OCR ve belge ayrıştırma modellerinin yakında yayınlanması da dahil olmak üzere modellerimizi genişletmeye devam edeceğiz. PaddleOCR 3.0'ı faydalı bulur veya projelerinizde kullanmak isterseniz, lütfen bu teknik raporu kaynak gösteriniz.

Referanslar

- J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. Gpt-4 teknik raporu. [arXiv preprint arXiv:2303.08774](#), 2023.
- R. AI. Rolmocr: Daha hızlı, daha hafif açık kaynaklı bir OCR modeli, 2025.
- Baidu-ERNIE-Takımı. Ernie 4.5 teknik raporu, 2025.
- L. Blecher, G. Cucurull, T. Scialom ve R. Stojnic. Nougat: Akademik belgeler için nöral optik anlama, 2023.

- breezedeus. Pix2text. <https://github.com/breezedeus/Pix2Text>, 2022. Accessed: 2025-06-23.
- R. Casey and E. Lecolinet. A survey of methods and strategies in character segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):690–706, 1996. doi: 10.1109/34.506792.
- Z. Chen, W. Wang, Y. Cao, Y. Liu, Z. Gao, E. Cui, J. Zhu, S. Ye, H. Tian, Z. Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.
- C. Cui, T. Gao, S. Wei, Y. Du, R. Guo, S. Dong, B. Lu, Y. Zhou, X. Lv, Q. Liu, X. Hu, D. Yu, and Y. Ma. Pp-lcnet: A lightweight cpu convolutional neural network, 2021. URL <https://arxiv.org/abs/2109.15099>.
- Docling Team. Docling. <https://github.com/docling-project/docling>, 2024. Accessed: 2025-06-23.
- Y. Du, C. Li, R. Guo, X. Yin, W. Liu, J. Zhou, Y. Bai, Z. Yu, Y. Yang, Q. Dang, et al. Pp-ocr: A practical ultra lightweight ocr system. *arXiv preprint arXiv:2009.09941*, 2020.
- Y. Du, C. Li, R. Guo, C. Cui, W. Liu, J. Zhou, B. Lu, Y. Yang, Q. Liu, X. Hu, et al. Pp-ocrv2: Bag of tricks for ultra lightweight ocr system. *arXiv preprint arXiv:2109.03144*, 2021.
- Y. Du, Z. Chen, C. Jia, X. Yin, T. Zheng, C. Li, Y. Du, and Y.-G. Jiang. Svtr: Scene text recognition with a single visual model. *arXiv preprint arXiv:2205.00159*, 2022.
- Filimoa. open-parse. <https://github.com/Filimoa/open-parse>, 2024. Accessed: 2025-06-23.
- I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet. Multi-digit number recognition from street view imagery using deep convolutional neural networks, 2014. URL <https://arxiv.org/abs/1312.6082>.
- D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- J. Ha, R. M. Haralick, and I. T. Phillips. Recursive xy cut using bounding boxes of connected components. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 2, pages 952–955. IEEE, 1995.
- hiroisora. Umi-ocr. <https://github.com/hiroisora/Umi-OCR>, 2022. Accessed: 2025-06-23.
- W. Hu, X. Cai, J. Hou, S. Yi, and Z. Lin. Gtc: Guided training of ctc towards efficient and accurate scene text recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11005–11012, 2020.
- Intel Corporation. OpenVINO Toolkit. <https://www.intel.com/content/www/us/en/developer/tools/openvino-toolkit/overview.html>, 2018. Accessed: 2025-06-23.
- KevinHuSh. ragflow. <https://github.com/infiniflow/ragflow>, 2023. Accessed: 2025-06-23.
- breezedeus. Pix2text. <https://github.com/breezedeus/Pix2Text>, 2022. Erişim tarihi : 2025-06-23.
- R. Casey ve E. Lecolinet. Karakter segmentasyonunda yöntemler ve stratejiler üzerine bir anket. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):690–706, 1996. doi: 10.1109/34.506792.
- Z. Chen, W. Wang, Y. Cao, Y. Liu, Z. Gao, E. Cui, J. Zhu, S. Ye, H. Tian, Z. Liu, et al. Model, veri ve test zamanı ölçeklendirme ile açık kaynaklı çok modlu modellerin performans sınırlarını genişletme. *arXiv preprint arXiv:2412.05271*, 2024.
- C. Cui, T. Gao, S. Wei, Y. Du, R. Guo, S. Dong, B. Lu, Y. Zhou, X. Lv, Q. Liu, X. Hu, D. Yu ve Y. Ma. Pp-lcnet: Hafif bir CPU evrişimsel sınır ağı, 2021. URL <https://arxiv.org/abs/2109.15099>.
- Docling Ekibi. Docling. <https://github.com/docling-project/docling>, 2024. Erişim tarihi: 2025-06-23.
- Y. Du, C. Li, R. Guo, X. Yin, W. Liu, J. Zhou, Y. Bai, Z. Yu, Y. Yang, Q. Dang, et al. Pp-ocr: Pratik ultra hafif bir OCR sistemi. *arXiv preprint arXiv:2009.09941*, 2020.
- Y. Du, C. Li, R. Guo, C. Cui, W. Liu, J. Zhou, B. Lu, Y. Yang, Q. Liu, X. Hu, et al. Pp-ocrv2: Ultra hafif OCR sistemi için püf noktaları. *arXiv preprint arXiv:2109.03144*, 2021.
- Y. Du, Z. Chen, C. Jia, X. Yin, T. Zheng, C. Li, Y. Du ve Y.-G. Jiang. Svtr: Tek bir görsel modelle sahne metni tanıma. *arXiv ön baskısı arXiv:2205.00159*, 2022.
- Filimoa. açık ayrıştırma. <https://github.com/Filimoa/open-parse>, 2024. Erişim tarihi : 2025-06-23.
- I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud ve V. Shet. Derin evrişimli sınır ağları kullanarak sokak görünümü görüntülerinden çok haneli sayı tanıma, 2014. URL <https://arxiv.org/abs/1312.6082>.
- D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, vd. Deepseek-r1: Takviyeli öğrenme yoluyla LLM'lerde muhakeme yeteneğini teşvik etme. *arXiv ön baskısı arXiv:2501.12948*, 2025.
- J. Ha, R. M. Haralick ve I. T. Phillips. Bağlantılı bileşenlerin sınırlayıcı kutularını kullanarak özyinelemeli xy kesimi. 3. Uluslararası Belge Analizi ve Tanıma Konferansı Bildirileri, cilt 2 , sayfa 952–955. IEEE, 1995.
- hiroisora. Umi-ocr. <https://github.com/hiroisora/Umi-OCR>, 2022. Erişim tarihi: 2025-06-23.
- W. Hu, X. Cai, J. Hou, S. Yi ve Z. Lin. Gtc: Verimli ve doğru sahne metni tanımayaya yönelik ctc 'nin rehberli eğitimi. *AAAI Yapay Zeka Konferansı Bildirileri*, cilt 34, sayfa 11005–11012, 2020 .
- Intel Corporation. OpenVINO Toolkit. <https://www.intel.com/content/www/us/en/developer/tools/openvino-toolkit/overview.html>, 2018. Erişim tarihi: 2025-06-23.
- KevinHuSh. ragflow. <https://github.com/infiniflow/ragflow> , 2023. Erişim tarihi : 2025-06-23.

- P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.
- C. Li, R. Guo, J. Zhou, M. An, Y. Du, L. Zhu, Y. Liu, X. Hu, and D. Yu. Pp-structurev2: A stronger document analysis system. *arXiv preprint arXiv:2210.05391*, 2022a.
- C. Li, W. Liu, R. Guo, X. Yin, K. Jiang, Y. Du, Y. Du, L. Zhu, B. Lai, X. Hu, et al. Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. *arXiv preprint arXiv:2206.03001*, 2022b.
- C. Li, W. Liu, R. Guo, X. Yin, K. Jiang, Y. Du, Y. Du, L. Zhu, B. Lai, X. Hu, et al. Pp-ocrv3: More attempts for the improvement of ultra lightweight ocr system. *arXiv preprint arXiv:2206.03001*, 2022c.
- H. Liu, C. Cui, Y. Du, Y. Liu, and G. Pan. Pp-formulanet: Bridging accuracy and efficiency in advanced formula recognition. *arXiv preprint arXiv:2503.18382*, 2025.
- Y. Ma, D. Yu, T. Wu, and H. Wang. Paddlepaddle: An open-source deep learning platform from industrial practice. *Frontiers of Data and Domputing*, 1(1):105–115, 2019.
- Microsoft Corporation. ONNX Runtime. <https://github.com/microsoft/onnxruntime>, 2018. Accessed: 2025-06-23.
- S. Mori, H. Nishida, and H. Yamada. *Optical Character Recognition*. John Wiley & Sons, 1999.
- A. Nassar, A. Marafioti, M. Omenetti, M. Lysak, N. Livathinos, C. Auer, L. Morin, R. T. de Lima, Y. Kim, A. S. Gurbuz, et al. Smoldocling: An ultra-compact vision-language model for end-to-end multi-modal document conversion. *arXiv preprint arXiv:2503.11576*, 2025.
- F. Ni, K. Huang, Y. Lu, W. Lv, G. Wang, Z. Chen, and Y. Liu. Pp-docbee: Improving multimodal document understanding through a bag of tricks. *arXiv preprint arXiv:2503.04065*, 2025.
- NVIDIA Corporation. TensorRT. <https://developer.nvidia.com/tensorrt>, 2017. Accessed: 2025-06-23.
- NVIDIA Corporation. Triton Inference Server. <https://github.com/triton-inference-server/server>, 2018. Accessed: 2025-06-23.
- L. Ouyang, Y. Qu, H. Zhou, J. Zhu, R. Zhang, Q. Lin, B. Wang, Z. Zhao, M. Jiang, X. Zhao, et al. Omnidocbench: Benchmarking diverse pdf document parsing with comprehensive annotations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24838–24848, 2025.
- PaddlePaddle Team. Ai studio. <https://aistudio.baidu.com>, 2019. Accessed: 2025-06-23.
- V. Paruchuri. Marker. <https://github.com/VikParuchuri/marker>, 2023. Accessed: 2025-06-23.
- J. Poznanski, J. Borchardt, J. Dunkelberger, R. Huff, D. Lin, A. Rangapur, C. Wilhelm, K. Lo, and L. Soldaini. olmocr: Unlocking trillions of tokens in pdfs with vision language models. *arXiv preprint arXiv:2502.18443*, 2025.
- S. Ramírez. FastAPI. <https://github.com/fastapi/fastapi>, 2018. Accessed: 2025-06-23.
- P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al. Bilgi yoğun NLP görevleri için erişim artırılmış üretim. *Advances in neural information processing systems*, 33:9459–9474, 2020.
- C. Li, R. Guo, J. Zhou, M. An, Y. Du, L. Zhu, Y. Liu, X. Hu, and D. Yu. Pp-structurev2: Daha güçlü bir belge analiz sistemi. *arXiv preprint arXiv:2210.05391*, 2022a.
- C. Li, W. Liu, R. Guo, X. Yin, K. Jiang, Y. Du, Y. Du, L. Zhu, B. Lai, X. Hu, et al. Pp-ocrv3: Ultra hafif bir OCR sisteminin iyileştirilmesi için daha fazla deneme. *arXiv ön baskı arXiv:2206.03001*, 2022b.
- C. Li, W. Liu, R. Guo, X. Yin, K. Jiang, Y. Du, Y. Du, L. Zhu, B. Lai, X. Hu, et al. Pp-ocrv3: Ultra hafif bir OCR sisteminin iyileştirilmesi için daha fazla deneme. *arXiv ön baskı arXiv:2206.03001*, 2022c.
- H. Liu, C. Cui, Y. Du, Y. Liu ve G. Pan. Pp-formulanet: Gelişmiş formül tanımada doğruluk ve verimlilik arasında köprü kurmak. *arXiv ön baskı arXiv:2503.18382*, 2025.
- Y. Ma, D. Yu, T. Wu ve H. Wang. Paddlepaddle: Endüstriyel uygulamalardan açık kaynaklı bir derin öğrenme platformu. *Frontiers of Data and Domputing*, 1(1):105–115, 2019.
- Microsoft Corporation. ONNX Runtime. <https://github.com/microsoft/onnxruntime>, 2018. Erişim tarihi: 2025-06-23.
- S. Mori, H. Nishida ve H. Yamada. *Optik Karakter Tanıma*. John Wiley & Sons, 1999.
- A. Nassar, A. Marafioti, M. Omenetti, M. Lysak, N. Livathinos, C. Auer, L. Morin, R. T. de Lima, Y. Kim, A. S. Gurbuz, et al. Smoldocling: Uçtan uca çok modlu belge dönüşümü için ultra kompakt bir görüş-dil modeli. *arXiv ön baskısı arXiv:2503.11576*, 2025.
- F. Ni, K. Huang, Y. Lu, W. Lv, G. Wang, Z. Chen ve Y. Liu. Pp-docbee: Bir dizi teknikle çok modlu belge anlaması yeteneğini geliştirmek. *arXiv ön baskısı arXiv:2503.04065*, 2025.
- NVIDIA Corporation. TensorRT. <https://developer.nvidia.com/tensorrt>, 2017. Erişim Tarihi: 2025-06-23.
- NVIDIA Corporation. Triton Inference Server. <https://github.com/triton-inference-server/server>, 2018. Erişim tarihi: 2025-06-23.
- L. Ouyang, Y. Qu, H. Zhou, J. Zhu, R. Zhang, Q. Lin, B. Wang, Z. Zhao, M. Jiang, X. Zhao, et al. Omnidocbench: Kapsamlı notasyonlarla çeşitli PDF belge ayırtırma karşılaştırması. *Computer Vision and Pattern Recognition Conference Bildirileri*, sayfa 24838–24848, 2025.
- PaddlePaddle Ekibi. Ai studio. <https://aistudio.baidu.com>, 2019. Erişim tarihi: 2025-06-23.
- V. Paruchuri. Marker. <https://github.com/VikParuchuri/marker>, 2023. Erişim tarihi: 2025-06-23.
- J. Poznanski, J. Borchardt, J. Dunkelberger, R. Huff, D. Lin, A. Rangapur, C. Wilhelm, K. Lo ve L. Soldaini. olmocr: Görme dil modelleriyle PDF'lerdeki trilyonlarca token'i açığa çıkarmak. *arXiv ön baskısı arXiv:2502.18443*, 2025.
- S. Ramírez. FastAPI. <https://github.com/fastapi/fastapi>, 2018. Erişim tarihi: 2025-06-23.

- B. Shi, X. Bai, and C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition, 2015. URL <https://arxiv.org/abs/1507.05717>.
- T. Sun, C. Cui, Y. Du, and Y. Liu. Pp-doclayout: A unified document layout detection model to accelerate large-scale data construction. arXiv preprint arXiv:2503.17213, 2025.
- Unstructured-IO. unstructured. <https://github.com/Unstructured-IO/unstructured>, 2022. Accessed: 2025-06-23.
- F. Verhoeven, T. Magne, and O. Sorkine-Hornung. Uvdoc: neural grid-based document unwarping. In SIGGRAPH Asia 2023 Conference Papers, pages 1–11, 2023.
- B. Wang, C. Xu, X. Zhao, L. Ouyang, F. Wu, Z. Zhao, R. Xu, K. Liu, Y. Qu, F. Shang, et al. Mineru: An open-source solution for precise document content extraction. arXiv preprint arXiv:2409.18839, 2024.
- H. Wei, C. Liu, J. Chen, J. Wang, L. Kong, Y. Xu, Z. Ge, L. Zhao, J. Sun, Y. Peng, et al. General ocr theory: Towards ocr-2.0 via a unified end-to-end model. 2024.
- A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang, and Z. Qiu. Qwen2.5 technical report. arXiv preprint arXiv:2412.15115, 2024.
- A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. Qwen3 technical report. arXiv preprint arXiv:2505.09388, 2025.
- B. Shi, X. Bai ve C. Yao. Görüntü tabanlı dizi tanıma için uçtan uca eğitilebilir bir sinir ağı ve bunun sahne metni tanımaya uygulanması, 2015. URL <https://arxiv.org/abs/1507.05717>.
- T. Sun, C. Cui, Y. Du ve Y. Liu. Pp-doclayout: Büyük ölçekli veri oluşturmayı hızlandırmak için birleşik bir belge düzeni algılama modeli. arXiv ön baskısı arXiv:2503.17213, 2025.
- Unstructured-IO. yapılandırılmamış. <https://github.com/Unstructured-IO/unstructured>, 2022. Erişim tarihi: 2025-06-23.
- F. Verhoeven, T. Magne ve O. Sorkine-Hornung. Uvdoc: sinir ağı tabanlı belge düzeltme. SIGGRAPH Asya 2023 Konferans Bildirileri, sayfa 1–11, 2023.
- B. Wang, C. Xu, X. Zhao, L. Ouyang, F. Wu, Z. Zhao, R. Xu, K. Liu, Y. Qu, F. Shang, vd. Mineru: Hassas belge içeriği çıkarımı için açık kaynaklı bir çözüm. arXiv ön baskısı arXiv:2409.18839, 2024.
- H. Wei, C. Liu, J. Chen, J. Wang, L. Kong, Y. Xu, Z. Ge, L. Zhao, J. Sun, Y. Peng, vd. Genel OCR teorisi: Birleşik uçtan uca model aracılığıyla OCR-2.0'a doğru. 2024.
- A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei, H. Lin, J. Yang, J. Tu, J. Zhang, J. Yang, J. Yang, J. Zhou, J. Lin, K. Dang, K. Lu, K. Bao, K. Yang, L. Yu, M. Li, M. Xue, P. Zhang, Q. Zhu, R. Men, R. Lin, T. Li, T. Xia, X. Ren, X. Ren, Y. Fan, Y. Su, Y. Zhang, Y. Wan, Y. Liu, Z. Cui, Z. Zhang ve Z. Qiu. Qwen2.5 teknik raporu. arXiv ön baskı arXiv:2412.15115, 2024.
- A. Yang, A. Li, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Gao, C. Huang, C. Lv, et al. Qwen3 teknik raporu. arXiv ön baskısı arXiv:2505.09388, 2025.

Appendix

A. Acknowledgments

We gratefully acknowledge all individuals who supported this work through their invaluable contributions to data construction, deployment, testing, project maintenance, product development, online demo creation, and operations. Their dedication and efforts have played a crucial role in the successful advancement and ongoing improvement of this project.

Baoku Yu	Jiahua Wang	Siyu Cheng	Yiqiao Zhou
Chang Xu	Jianying Qu	Suyin Liang	Ye Han
Chao Han	Jiaxin Sui	Tao Luo	Yongkun Du
Chunli Xie	Jinghui Duan	Tianyu Zheng	Zewu Wu
Guanzhong Wang	JingsongLiu	Xiaolong Ma	Zeyu Luo
Haitao Yu	Mengmeng Guo	Xin Li	Zhe Wang
Hengxin Chen	Min Zhuang	Xin Wang	Zhongkai Sun
Hong Cheng	Runlong Li	Xinran Liu	
Jiahao Bai	Shengjian Guo	Yimin Gao	

We would also like to acknowledge the invaluable contributions of open-source developers on GitHub, including but not limited to [@timminator](#), [@ackinc](#), [@Appla](#), [@co63oc](#), [@jk4e](#), and many others whose work has inspired and supported our project.

Finally, we express our sincere gratitude to all project contributors for their long-term support and valuable input. Their efforts have greatly advanced the development and improvement of this project. Figure 12 shows the avatars of these contributors, as collected from the PaddleOCR GitHub repository.

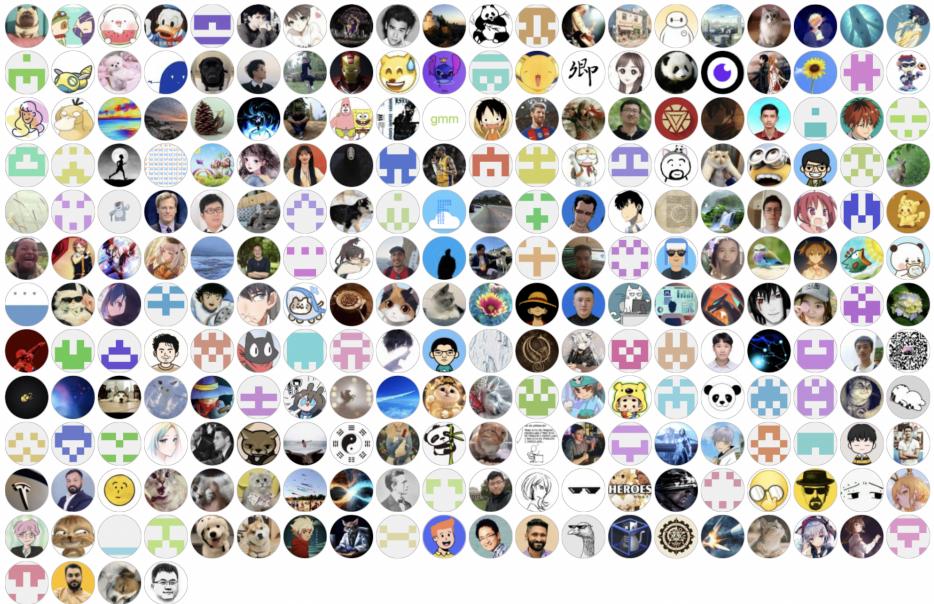


Figure 12 | Avatars of long-term contributors to the PaddleOCR project.

Ek

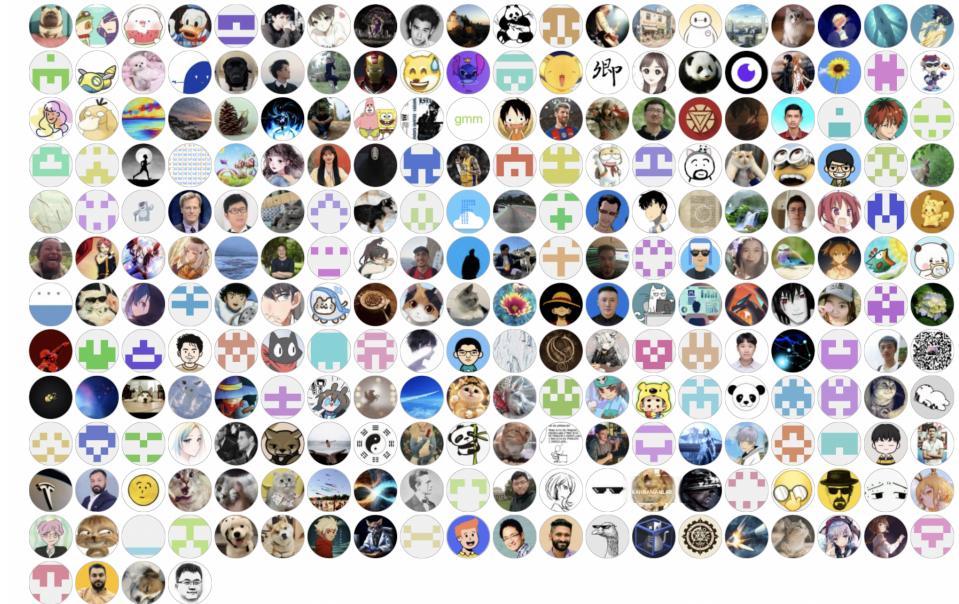
A. Teşekkürler

Veri oluşturma, dağıtım, test etme, proje bakımı, ürün geliştirme, çevirmişi demo oluşturma ve operasyonlara paha biçilmez katkılarıyla bu çalışmayı destekleyen tüm kişilere minnettarız. Adanmışlıklarını ve çabaları, bu projenin başarılı ilerlemesinde ve sürekli iyileştirilmesinde kritik bir rol oynamıştır.

Baoku Yu	Jiahua Wang	Siyu Cheng	Yiqiao Zhou
Chang Xu	Jianying Qu	Suyin Liang	Ye Han
Chao Han	Jiaxin Sui	Tao Luo	Yongkun Du
Chunli Xie	Jinghui Duan	Tianyu Zheng	Zewu Wu
Guanzhong Wang	JingsongLiu	Xiaolong Ma	Zeyu Luo
Haitao Yu	Mengmeng Guo	Xin Li	Zhe Wang
Hengxin Chen	Min Zhuang	Xin Wang	Zhongkai Sun
Hong Cheng	Runlong Li	Xinran Liu	
Jiahao Bai	Shengjian Guo	Yimin Gao	

Ayrıca, [@timminator](#), [@ackinc](#), [@Appla](#), [@co63oc](#), [@jk4e](#) ve çalışmaları projemize ilham veren ve destekleyen diğer pek çok kişinin de aralarında bulunduğu GitHub'daki açık kaynak geliştiricilerin paha biçilmez katkılarını takdir etmek isteriz.

Son olarak, tüm proje katkıda bulunanlarına uzun vadeli destekleri ve değerli girdileri için içten şükranlarımızı sunarız. Bu çabalar, projenin geliştirilmesini ve iyileştirilmesini büyük ölçüde ilerletmiştir. Şekil 12, PaddleOCR GitHub deposundan toplanan bu katkıda bulunanların avatarlarını göstermektedir.



Şekil 12 | PaddleOCR projesine uzun vadeli katkıda bulunanların avatarları.

B. Usage of command and API details

To use PaddleOCR 3.0, you can simply install the paddleocr package from PyPI. PaddleOCR 3.0 provides command-line interface (CLI) and Python API for users to use conveniently.

B.1. Run inference by CLI

We provide convenient CLI methods for users to quickly experience the capabilities of PP-OCRv5, PP-StructureV3, and PP-ChatOCRv4, as follows:

```
1 # Run PP-OCRv5 inference
2 paddleocr ocr -i test.png \
3     --use_doc_orientation_classify False \
4     --use_doc_unwarping False \
5     --use_textline_orientation False
6
7 # Run PP-StructureV3 inference
8 paddleocr pp_structurev3 -i test.png \
9     --use_doc_orientation_classify False \
10    --use_doc_unwarping False
11
12 # Get the Qianfan API Key at first, then run PP-ChatOCRv4
13 paddleocr pp_chatocrv4_doc -i test.png \
14     -k number \
15     --qianfan_api_key your_api_key \
16     --use_doc_orientation_classify False \
17     --use_doc_unwarping False
```

B.2. Run inference by Python API

We also provide a clean interface to facilitate users in using and integrating it into their own projects.

1. PP-OCRv5 Example

```
1 # Initialize PaddleOCR instance
2 from paddleocr import PaddleOCR
3 ocr = PaddleOCR(
4     use_doc_orientation_classify=False,
5     use_doc_unwarping=False,
6     use_textline_orientation=False)
7
8 # Run OCR inference on a sample image
9 result = ocr.predict(input="test.png")
10
11 # Visualize the results and save the JSON results
12 for res in result:
13     res.print()
14     res.save_to_img("output")
15     res.save_to_json("output")
```

2. PP-StructureV3 Example

```
1 # Initialize PPStructureV3 instance
2 from paddleocr import PPStructureV3
3
4 pipeline = PPStructureV3(
5     use_doc_orientation_classify=False,
6     use_doc_unwarping=False)
7
8 # Run PPStructureV3 inference
9 output = pipeline.predict(input="test.png")
```

B. Komut ve API Detaylarının Kullanımı

PaddleOCR 3.0'ı kullanmak için, paddleocr paketini PyPI'dan kolayca kurabilirsiniz. PaddleOCR 3.0, kullanıcıların rahatlıkla kullanabilmeleri için komut satırı arayüzü (CLI) ve Python API'si sunmaktadır.

B.1. CLI ile Çıkarım Çalıştırma

Kullanıcıların PP-OCRv5, PP-StructureV3 ve PP-ChatOCRv4 yeteneklerini hızlıca deneyimlemeleri için aşağıdaki gibi kullanışlı CLI yöntemleri sunuyoruz:

```
1 # PP-OCRv5 Çıkarımı Çalıştırma
2 paddleocr ocr -i test.png \
3     --use_doc_orientation_classify False \
4     --use_doc_unwarping False \
5     --use_textline_orientation False
6
7 # PP-StructureV3 Çıkarımı Çalıştırma
8 paddleocr pp_structurev3 -i test.png \
9     --use_doc_orientation_classify False \
10    --use_doc_unwarping False
11
12 # Öncelikle Qianfan API Anahtarını alın, ardından PP-ChatOCRv4'ü çalıştırın.
13 paddleocr pp_chatocrv4_doc -i test.png \
14     -k number \
15     --qianfan_api_key your_api_key \
16     --use_doc_orientation_classify False \
17     --use_doc_unwarping False
```

B.2. Python API ile Çıkarım Çalıştırma

Kullanıcıların kendi projelerinde kullanımını ve entegrasyonunu kolaylaştırmak için temiz bir arayüz de sağlıyoruz.

1. PP-OCRv5 Örneği

```
1 # PaddleOCR Örneğini Başlatma
2 from paddleocr import PaddleOCR
3 ocr = PaddleOCR(
4     use_doc_orientation_classify=False,
5     use_doc_unwarping=False,
6     use_textline_orientation=False)
7
8 # Örnek Bir Görüntü Üzerinde OCR Çıkarımını Çalıştırma
9 result = ocr.predict(input="test.png")
10
11 # Sonuçları Görselleştirme ve JSON Sonuçlarını Kaydetme
12 for res in result:
13     res.print()
14     res.save_to_img("output")
15     res.save_to_json("output")
```

2. PP-StructureV3 Örneği

```
1 # PPStructureV3 örneğini başlat
2 from paddleocr import PPStructureV3
3
4 pipeline = PPStructureV3(
5     use_doc_orientation_classify=False,
6     use_doc_unwarping=False)
7
8 # PPStructureV3 çıkışımını çalıştır
9 output = pipeline.predict(input="test.png")
```

```

10 # Visualize the results and save the JSON results
11 for res in output:
12     res.print()
13     res.save_to_json(save_path="output")
14     res.save_to_markdown(save_path="output")
15

```

3. PP-ChatOCRv4 Example

```

1 from paddleocr import PPChatOCRv4Doc
2
3 chat_bot_config = {
4     "module_name": "chat_bot",
5     "model_name": "xxx",
6     "base_url": "https://qianfan.baidubce.com/v2",
7     "api_type": "openai",
8     "api_key": "api_key", # your api_key
9 }
10
11 retriever_config = {
12     "module_name": "retriever",
13     "model_name": "embedding-v1",
14     "base_url": "https://qianfan.baidubce.com/v2",
15     "api_type": "qianfan",
16     "api_key": "api_key", # your api_key
17 }
18
19 # Initialize PPChatOCRv4Doc instance
20 pipeline = PPChatOCRv4Doc(
21     use_doc_orientation_classify=False,
22     use_doc_unwarping=False)
23
24 visual_predict_res = pipeline.visual_predict(
25     input="test.png",
26     use_common_ocr=True,
27     use_seal_recognition=True,
28     use_table_recognition=True)
29
30 mllm_predict_info = None
31 use_mllm = False
32 visual_info_list = []
33
34 for res in visual_predict_res:
35     visual_info_list.append(res["visual_info"])
36     layout_parsing_result = res["layout_parsing_result"]
37
38 vector_info = pipeline.build_vector(
39     visual_info_list,
40     flag_save_bytes_vector=True,
41     retriever_config=retriever_config)
42
43 chat_result = pipeline.chat(
44     key_list=[ "number of people" ],
45     visual_info=visual_info_list,
46     vector_info=vector_info,
47     mllm_predict_info=mllm_predict_info,
48     chat_bot_config=chat_bot_config,
49     retriever_config=retriever_config)
50 print(chat_result)

```

```

10 # Sonuçları Görselleştirme ve JSON Sonuçlarını Kaydetme
11 for res in output:
12     res.print()
13     res.save_to_json(save_path="output")
14     res.save_to_markdown(save_path="output")
15

```

3. PP-ChatOCRv4 Örneği

```

1 from paddleocr import PPChatOCRv4Doc
2
3 chat_bot_config = {
4     "modül_adi": "chat_bot",
5     "model_adi": "xxx",
6     "base_url": "https://qianfan.baidubce.com/v2",
7     "api_type": "openai",
8     "api_key": "api_key", # sizin api_key'iniz
9 }
10
11 retriever_config = {
12     "modül_adi": "retriever",
13     "model_adi": "embedding-v1",
14     "base_url": "https://qianfan.baidubce.com/v2",
15     "api_type": "qianfan",
16     "api_key": "api_key", # sizin api_key'iniz
17 }
18
19 # PPChatOCRv4Doc örneğini başlat
20 pipeline = PPChatOCRv4Doc(
21     use_doc_orientation_classify=False,
22     use_doc_unwarping=False)
23
24 visual_predict_res = pipeline.visual_predict(
25     input="test.png",
26     use_common_ocr=True,
27     use_seal_recognition=True,
28     use_table_recognition=True)
29
30 mllm_predict_info = None
31 use_mllm = False
32 visual_info_list = []
33
34 for res in visual_predict_res:
35     visual_info_list.append(res["visual_info"])
36     layout_parsing_result = res["layout_parsing_result"]
37
38 vector_info = pipeline.build_vector(
39     visual_info_list,
40     flag_save_bytes_vector=True,
41     retriever_config=retriever_config)
42
43 chat_result = pipeline.chat(
44     key_list=[ "number of people" ],
45     visual_info=visual_info_list,
46     vector_info=vector_info,
47     mllm_predict_info=mllm_predict_info,
48     chat_bot_config=chat_bot_config,
49     retriever_config=retriever_config)
50 print(chat_result)

```

C. More details on MCP host configuration

Here are several example configurations for the MCP host in Claude for Desktop, illustrating how to connect to a PaddleOCR MCP server. Each configurable parameter may be specified either via environment variables (as shown in the env field) or via command-line arguments (as shown in the args field).

1. Using the local Python library:

```
1 {
2     "mcpServers": {
3         "paddleocr-ocr": {
4             "command": "paddleocr_mcp",
5             "args": [
6                 "--device", "gpu:1"
7             ],
8             "env": {
9                 "PADDLEOCR_MCP_PIPELINE": "OCR",
10                "PADDLEOCR_MCP_PPOCR_SOURCE": "local"
11            }
12        }
13    }
14 }
```

2. Using a PaddlePaddle AI Studio service:

```
1 {
2     "mcpServers": {
3         "paddleocr-ocr": {
4             "command": "paddleocr_mcp",
5             "args": [
6                 "--timeout", "60"
7             ],
8             "env": {
9                 "PADDLEOCR_MCP_PIPELINE": "OCR",
10                "PADDLEOCR_MCP_PPOCR_SOURCE": "aistudio",
11                "PADDLEOCR_MCP_SERVER_URL": "<server-url>",
12                "PADDLEOCR_MCP_AISTUDIO_ACCESS_TOKEN": "<access-token>"
13            }
14        }
15    }
16 }
```

3. Using a self-hosted service:

```
1 {
2     "mcpServers": {
3         "paddleocr-ocr": {
4             "command": "paddleocr_mcp",
5             "args": [],
6             "env": {
7                 "PADDLEOCR_MCP_PIPELINE": "OCR",
8                 "PADDLEOCR_MCP_PPOCR_SOURCE": "self_hosted",
9                 "PADDLEOCR_MCP_SERVER_URL": "<server-url>"
10            }
11        }
12    }
13 }
```

C. MCP ana bilgisayar yapılandırması hakkında daha fazla ayrıntı

İşte Claude for Desktop'taki MCP ana bilgisayarı için, bir PaddleOCR MCP sunucusuna nasıl bağlanılacağını gösteren birkaç örnek yapılandırma. Her yapılandırılabilir parametre, ortam değişkenleri aracılığıyla (env alanında gösterildiği gibi) veya komut satırı argümanları aracılığıyla (args alanında gösterildiği gibi) belirtilebilir.

1. Yerel Python kütüphanesini kullanarak:

```
1 {
2     "mcpServers": {
3         "paddleocr-ocr": {
4             "command": "paddleocr_mcp",
5             "args": [
6                 "--device", "gpu:1"
7             ],
8             "env": {
9                 "PADDLEOCR_MCP_PIPELINE": "OCR",
10                "PADDLEOCR_MCP_PPOCR_SOURCE": "local"
11            }
12        }
13    }
14 }
```

2. PaddlePaddle AI Studio hizmetini kullanarak:

```
1 {
2     "mcpServers": {
3         "paddleocr-ocr": {
4             "command": "paddleocr_mcp",
5             "args": [
6                 "--timeout", "60"
7             ],
8             "env": {
9                 "PADDLEOCR_MCP_PIPELINE": "OCR",
10                "PADDLEOCR_MCP_PPOCR_SOURCE": "aistudio",
11                "PADDLEOCR_MCP_SERVER_URL": "<server-url>",
12                "PADDLEOCR_MCP_AISTUDIO_ACCESS_TOKEN": "<access-token>"
13            }
14        }
15    }
16 }
```

3. Kendi barındırılan hizmeti kullanarak:

```
1 {
2     "mcpServers": {
3         "paddleocr-ocr": {
4             "command": "paddleocr_mcp",
5             "args": [],
6             "env": {
7                 "PADDLEOCR_MCP_PIPELINE": "OCR",
8                 "PADDLEOCR_MCP_PPOCR_SOURCE": "self_hosted",
9                 "PADDLEOCR_MCP_SERVER_URL": "<sunucu-url>"
10            }
11        }
12    }
13 }
```