

September 19, 2025

mkdir output

index.html

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8" />
  <title>OCR ツール (画像・PDF対応) </title>
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <style>
    * {
      box-sizing: border-box;
    }
    body {
      font-family: 'Arial', 'Noto Sans JP', sans-serif;
      background: #f5f5f5;
      margin: 0;
      padding: 20px;
    }
    .container {
      max-width: 700px;
      margin: auto;
      background: white;
      padding: 30px;
      border-radius: 10px;
      box-shadow: 0 4px 12px rgba(0,0,0,0.1);
    }
    h1 {
      text-align: center;
      color: #007acc;
      margin-bottom: 30px;
      font-size: 1.5rem;
    }
    #dropArea {
      padding: 30px;
      border: 2px dashed #888;
      background-color: #f9f9f9;
      margin-bottom: 15px;
      text-align: center;
      color: #555;
      cursor: pointer;
      transition: background-color 0.3s;
      user-select: none;
    }
    #dropArea:hover {
      background-color: #eef;
    }
    .file-item {
      padding: 10px;
      border: 1px solid #ccc;
      margin: 6px 0;
      background: #fff;
      cursor: move;
      border-radius: 5px;
      display: flex;
      justify-content: space-between;
      align-items: center;
    }
    .file-name {
      overflow: hidden;
      text-overflow: ellipsis;
```

```

        white-space: nowrap;
        max-width: 90%;
    }
    .remove-btn {
        background: #e74c3c;
        border: none;
        color: white;
        font-weight: bold;
        border-radius: 3px;
        cursor: pointer;
        padding: 0 6px;
        font-size: 14px;
    }
    .sortable-ghost {
        opacity: 0.4;
    }
    input[type="file"],
    input[type="text"],
    button {
        width: 100%;
        font-size: 1rem;
        padding: 10px;
        margin: 10px 0;
        border-radius: 5px;
        border: 1px solid #ccc;
    }
    input[type="text"] {
        font-family: monospace;
    }
    button {
        background-color: #007acc;
        color: white;
        border: none;
        cursor: pointer;
        font-weight: bold;
        transition: background-color 0.2s;
    }
    button:hover {
        background-color: #005f99;
    }
    label {
        font-weight: bold;
        display: block;
        margin-top: 20px;
    }
</style>
</head>
<body>

<div class="container">
    <h1>画像・PDFからテキスト抽出（OCR）</h1>

    <form id="ocrForm" action="upload.php" method="post" enctype="multipart/form-data">
        <div id="dropArea">ここに画像（jpg/jpeg/png）またはPDFをドロップ、またはクリックして選択</div>
        <input type="file" id="fileSelector" name="file[]" accept=".jpg,.jpeg,.png,.pdf" multiple
            style="display: none;" required>

        <div id="fileList"></div>

        <label for="langInput">言語（tesseract対応言語コード）：</label>
        <input type="text" name="lang" id="langInput" value="jpn" required>

        <button type="submit">アップロードして抽出</button>
        言語は日本語 -> jpn<br>
        英語 -> eng
    </form>
</div>

<!-- SortableJS CDN -->
<script src="https://cdn.jsdelivr.net/npm/sortablejs@1.15.0/Sortable.min.js"></script>

```

```

<script>
  const dropArea = document.getElementById('dropArea');
  const fileList = document.getElementById('fileList');
  const fileSelector = document.getElementById('fileSelector');
  const form = document.getElementById('ocrForm');
  let files = [];

  function createFileItem(file) {
    const div = document.createElement('div');
    div.className = 'file-item';
    div.setAttribute('data-filename', file.name);

    const span = document.createElement('span');
    span.className = 'file-name';
    span.textContent = file.name;

    const btn = document.createElement('button');
    btn.className = 'remove-btn';
    btn.type = 'button';
    btn.textContent = '×';
    btn.addEventListener('click', () => {
      files = files.filter(f => f.name !== file.name);
      div.remove();
      checkRequired();
    });

    div.appendChild(span);
    div.appendChild(btn);

    return div;
  }

  // ファイル追加
  function addFile(file) {
    const allowedTypes = ['application/pdf', 'image/jpeg', 'image/png'];
    if (!allowedTypes.includes(file.type)) return;
    if (files.some(f => f.name === file.name)) return;

    files.push(file);
    const fileItem = createFileItem(file);
    fileList.appendChild(fileItem);
    checkRequired();
  }

  // 必須チェック (ファイルがないとrequired通らないため対応)
  function checkRequired() {
    if (files.length === 0) {
      fileSelector.setAttribute('required', 'required');
    } else {
      fileSelector.removeAttribute('required');
    }
  }

  // ドラッグ&ドロップイベント
  ['dragenter', 'dragover', 'dragleave', 'drop'].forEach(eventName => {
    dropArea.addEventListener(eventName, e => {
      e.preventDefault();
      e.stopPropagation();
    });
  });

  dropArea.addEventListener('dragover', () => dropArea.style.backgroundColor = '#eef');
  dropArea.addEventListener('dragleave', () => dropArea.style.backgroundColor = '#f9f9f9');

  dropArea.addEventListener('drop', (e) => {
    dropArea.style.backgroundColor = '#f9f9f9';
    const droppedFiles = Array.from(e.dataTransfer.files);
    droppedFiles.forEach(addFile);
  });

  // クリックでファイル選択

```

```

dropArea.addEventListener('click', () => fileSelector.click());

fileSelector.addEventListener('change', () => {
  Array.from(fileSelector.files).forEach(addFile);
  fileSelector.value = '';
});

// 並べ替え
Sortable.create(fileList, {
  animation: 150,
  ghostClass: 'sortable-ghost'
});

// フォーム送信時にファイルリストの順番通りに送る
form.addEventListener('submit', e => {
  e.preventDefault();

  if(files.length === 0) {
    alert('ファイルを選択してください');
    return;
  }

  const formData = new FormData(form);
  // クリアされているinput[type=file]は使わずに、JS上のfiles配列からappendする
  formData.delete('file[]');
  const items = fileList.querySelectorAll('.file-item');
  items.forEach(item => {
    const name = item.getAttribute('data-filename');
    const file = files.find(f => f.name === name);
    if (file) {
      formData.append('file[]', file, file.name);
    }
  });

  fetch(form.action, {
    method: form.method,
    body: formData,
  })
  .then(res => res.text())
  .then(text => {
    document.body.innerHTML = text;
  })
  .catch(err => {
    alert('アップロードに失敗しました');
    console.error(err);
  });
});

checkRequired();
</script>

</body>
</html>

```

upload.php

```

<?php
$uploadDir = 'output/';
if (!file_exists($uploadDir)) {
  mkdir($uploadDir, 0777, true);
}

$lang = $_POST['lang'] ?? 'jpn';

if (!isset($_FILES['file'])) {
  die('ファイルがアップロードされていません。');
}

$results = [];

foreach ($_FILES['file']['error'] as $index => $error) {
  if ($error !== UPLOAD_ERR_OK) {

```

```

        $results[] = "ファイル #" . ($index + 1) . " のアップロードに失敗しました。";
        continue;
    }

    $tmpName = $_FILES['file']['tmp_name'][$index];
    $origName = $_FILES['file']['name'][$index];
    $ext = strtolower(pathinfo($origName, PATHINFO_EXTENSION));
    $baseName = uniqid('ocr_');

    $uploadedPath = $uploadDir . $baseName . '.' . $ext;

    if (!move_uploaded_file($tmpName, $uploadedPath)) {
        $results[] = "ファイル {$origName} の保存に失敗しました。";
        continue;
    }

    // PDFの場合は1ページ目だけPNGに変換
    if ($ext === 'pdf') {
        $imgPath = $uploadDir . $baseName . '.png';
        $convertCmd = "/usr/bin/convert -density 300 " . escapeshellarg($uploadedPath) . "[0] -
            quality 100 " . escapeshellarg($imgPath) . " 2>&1";
        exec($convertCmd, $convertOut, $convertRet);
        if ($convertRet !== 0) {
            $results[] = "PDFファイル {$origName} の画像変換に失敗しました。";
            continue;
        }
    } else {
        $imgPath = $uploadedPath;
    }

    // OCR実行
    $outputPath = $uploadDir . $baseName;
    $tessdataDir = '/usr/share/tesseract-ocr/5/tessdata/';
    $ocrCmd = "TESSDATA_PREFIX={$tessdataDir} /usr/bin/tesseract " . escapeshellarg($imgPath) . " "
        . escapeshellarg($outputPath) . " -l " . escapeshellarg($lang) . " 2>&1";
    exec($ocrCmd, $ocrOut, $ocrRet);
    if ($ocrRet !== 0) {
        $results[] = "ファイル {$origName} のOCR処理に失敗しました。";
        continue;
    }

    $txtFile = $outputPath . '.txt';
    if (!file_exists($txtFile)) {
        $results[] = "ファイル {$origName} のOCR結果ファイルが見つかりません。";
        continue;
    }

    $resultText = file_get_contents($txtFile);
    $results[] = "<h3>【{$origName}】の抽出テキスト</h3><pre>" . htmlspecialchars($resultText,
        ENT_QUOTES, 'UTF-8') . "</pre>";
}

// 結果表示
echo "<div style='max-width: 700px; margin: auto; padding: 20px; font-family: sans-serif;'>";
echo "<h1>OCR結果</h1>";
echo implode("<hr>", $results);
echo "<p><a href='index.html'>戻る</a></p>";
echo "</div>";

```

```

sudo apt install tesseract-ocr
sudo apt install ghostscript
sudo apt install tesseract-ocr-jpn

```