**1. Skin Lesion Classification for Melanoma Detection**

* **Objective:** Develop a model to classify skin lesions and detect melanoma or other skin cancers using dermoscopic images.
* **Tech Stack:**
  + **Model Development:** Use CNNs and transfer learning models like EfficientNet or MobileNet for classification.
  + **Preprocessing:** OpenCV for image preprocessing such as segmentation, noise reduction, and contrast enhancement.
  + **Web Application:** A web app using TensorFlow.js for real-time classification of skin lesions, with Node.js for managing user sessions and data storage.
* **Practical Application:** This tool can aid dermatologists and patients in early melanoma detection, increasing the chances of successful treatment.

**2.** **Tuberculosis Detection from Chest X-ray Images**

* **Objective:** Create a model that can detect tuberculosis from chest X-ray images with high accuracy.
* **Tech Stack:**
  + **Model Development:** Use CNNs and transfer learning with models like VGG16 or ResNet for classification.
  + **Preprocessing:** OpenCV for noise reduction, segmentation of lung regions, and contrast enhancement.
  + **Web Application:** A web-based platform using TensorFlow.js for real-time X-ray analysis, with Node.js for backend support and patient data management.
* **Practical Application:** This project can be used in remote or under-resourced areas to quickly screen patients for tuberculosis, improving access to diagnosis and care.

**3. Brain Tumor Classification from MRI Scans**

* **Objective:** Build a deep learning model to classify brain tumors into different types (e.g., glioma, meningioma) using MRI scans.
* **Tech Stack:**
  + **Model Development:** Use CNNs and transfer learning models like U-Net or DenseNet for tumor segmentation and classification.
  + **Preprocessing:** OpenCV for skull stripping, normalization, and segmentation of MRI images.
  + **Web Application:** A web app using TensorFlow.js to visualize MRI scans and provide real-time tumor classification, with Node.js for managing patient data and analytics.
* **Practical Application:** This tool can support radiologists and neurosurgeons in diagnosing and planning treatment for brain tumors.

**4. Lung Nodule Detection and Classification from CT Scans**

* **Objective:** Create a model to detect and classify lung nodules in CT scans, distinguishing between benign and malignant cases.
* **Tech Stack:**
  + **Model Development:** Use 3D CNNs or 2D CNNs with slice-wise processing and transfer learning for nodule detection and classification.
  + **Preprocessing:** OpenCV for slice extraction, normalization, and contrast enhancement of CT images.
  + **Web Application:** A web-based viewer using TensorFlow.js to display CT slices and mark detected nodules, with Node.js for backend support.
* **Practical Application:** This project can be used in lung cancer screening programs to improve early detection and reduce false positives.