Image Preprocessing

In [2]:

```
# Importing Libraries

from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [3]:

```
# Image Augmentation

train_datagen = ImageDataGenerator(rescale = 1./255, shear_range = 0.2,
zoom_range = 0.2, horizontal_flip = True)
test_datagen = ImageDataGenerator(rescale = 1./255)
```

In [4]:

```
# Loading train and test set

X_train =
train_datagen.flow_from_directory(r"C:\Users\Lenovo\Downloads\SI-
GuidedProject-322096-1664773219-main\Dataset\training_set", target_size =
(64, 64), batch_size = 32, class_mode = 'categorical')
X_test = test_datagen.flow_from_directory(r"C:\Users\Lenovo\Downloads\SI-
GuidedProject-322096-1664773219-main\Dataset\test_set", target_size =
(64, 64), batch_size = 32, class_mode = 'categorical')
```

```
Found 15750 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [5]:

```
# checking indices

X_train.class_indices
```

Out[5]:

```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Building

In [6]:

```
# Importing Libraries

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Flatten
```

In [7]:

```
# Initializing the Model

model = Sequential()
```

In [8]:

```
# Adding Convolution Layer
```

```
model.add(Convolution2D((32), (3,3), input_shape = (64, 64, 3),
activation = 'relu'))
```

In [9]:

```
# Adding Pooling Layer

model.add(MaxPooling2D(pool_size = (2, 2)))
```

In [10]:

```
# Adding Flatten Layer

model.add(Flatten())
```

In [11]:

```
# Adding Hidden Layer

model.add(Dense(units = 512, kernel_initializer = 'random_uniform',
activation = 'relu'))
```

In [12]:

```
# Adding Output Layer

model.add(Dense(units = 9, kernel_initializer = 'random_uniform',
activation = 'softmax'))
```

In [13]:

```
# Compile the model

model.compile(loss = 'categorical_crossentropy', optimizer = 'adam',
metrics = ['accuracy'])
```

In [ ]:

```
# Fiiting the model

model.fit_generator(X_train, steps_per_epoch = 24, epochs = 10,
validation_data = X_test, validation_steps = 40)
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_12688\1270027362.py:3: UserWar
ning: `Model.fit_generator` is deprecated and will be removed in a future v
ersion. Please use `Model.fit`, which supports generators.
  model.fit_generator(X_train, steps_per_epoch = 24, epochs = 10, validatio
n_data = X_test, validation_steps = 40)
Epoch 1/10
24/24 [==============================] - ETA: 0s - loss: 1.6133 - accuracy:
 0.4974
```

In [15]:

```
# Saving the model

model.save('aslpng1.h5')
```