Algorithm:
These k's are chosen based on hit and trial.
Do this for k in [1, 2, 3, 5, 10, 15, 20,23,25,27,29, 30, 50, 75, 100, 150, 170]

1.) Cross-Validation to find the threshold error for each person
    We have been given 6 images per person in the training set. We divided it into train and validation sets. Train set consists of 5 images and validation set consists of 1 image.
So there are 6 sets of train and validation set :
{1,2,3,4,5} {6} => error_1
{2,3,4,5,6} {1} => error_2
{3,4,5,6,1} {2} => error_3
{4,5,6,1,2} {3} => error_4
{5,6,1,2,3} {4} => error_5
{6,1,2,3,4} {5} => error_6

error_i is a list of difference of squared errors between validation image and trsin image. Each element in error_i represents the error when there is a correct recognition.
thresholding_person_specific = []
For j in length(error_i):
        thresholding_person_specific = max(error_1[j], error_2[j], error_3[j], error_4[j], error_5[j])

Now test phase starts:
for each person in the test set:
        for each person in the train set:
                # for images present in the database.
                if squarred error < threshold and it's label matches with the label in the train set.
                        then there is a match in the database.
                # for images not present in the database
                If squarred error < threshold of current peron in the train set.
                        then there is a mismatch i.e. it is falsely recognisiing the image.

Return the the value where correct_prediction is max or false positive is min.

Best False Negative is for k = 25, it is 41
Total number of correct prediction = 163 out of 208 images.

For detailed implementation of the algorithm, please visit the code :)