***Related works***

* In pattern recognition problems it is essential to measure distance or similarity. Given the following example :

String T : A C C T C G A G T

Pattern p: \_ \_ \_ \_ C G A ­\_ \_

Pattern P can be matched in String T by adding four empty spaces before the pattern and two after.

* Authors in (Yeh and Cheng 2008), use Levenshtein distance applied to images and videos to determine feature vectors.
* In (Amir et al. 2004), the authors propose a new distance for string matching, similar to Levenshtein distance, with K-Mismatches on the given string. This proposed approach was implemented with Message Passing Interface (MPI), and proved to be useful to establish similarity between strings.
* Authors in (Knuth et al. 1977), proposed an algorithm for pattern matching in strings, with running time proportional to the sum of the length of the strings. This traditional algorithm is now known as KMP string matching algorithm.
* Other classical string pattern matching algorithm was proposed in (Hussain et al. 2013), named Bidirectional Exact Pattern Matching (BDEPM). This algorithm introduces the idea to compare strings using pointers in simultaneous, one from the left other from the right.
* In (Alsmadi and Nuser 2012), they evaluated two algorithms for DNA string comparison in terms of accuracy and performance. The Longest Common Substring (LCS) algorithm, and Longest Common Sub-Sequence (LCSS) algorithms. In the following example, the highlighted letters, CTCT, in the sequences is LCSS of the specified sequences.
* Aho-Corasick and CommentZ-Walter algorithms (Vidanagamachchi et al. 2012) are two types of multiple patterns matching algorithms, authors in (Vidanagamachchi et al. 2012) implemented these two algorithms and worked with peptide sequences to study their accuracy and execution time. Results show that Aho-Corasick performs better than the CommentZ-Walter algorithm.

***Thank you,***