IAPR Workshop on CV - Special Hardware and Industrial Applications OCT.12-14, 1988, Tokyo

A New and Quick Approach To The Fault-Detection of High-Voltage Wires Based on Infrered Image Analysis

Xiabin Li Tsinghua University , Beijing

Guifang Zhen & Huazhi Liu

Huazhong University of Science & Technology , Wuhan

Abstract:
The analysis of the infrered images of high-voltage wires has been an effective technique in the fault detection of the high-tension network . In this paper, a new and simple approach to auto-detection is proposed . Featured by sample analysis , pattern recognition based on knowledge and context , and adaptive tracking as well , the new method can provide a quick and effective classification , and hence an auto-detection of the faults of the wires.

1 Introduction

Among all the faults and defects of the high voltage wires , one phenomenum is very common. It's the loosening of the joint of two wires. The improper connection gives rise to greater resistance , which produces much heat and wastes a lot of energy . As a result , the service life of the wires will be shortened . Mordern detection technique made much use of this heat radiation . The usual way is to analyse the wires's infrered pictures taken on a helicopter which flies along the high-tension wires . Since the improper connection point usually leaves a bright spot on the infrered pictures , it can possibly be detected by computers . In this paper , a new approach is proposed to make automatic detection . The new method uses only several rows of image elements to recognize the wire out of the background , consequently computations have been reduced greatly . A correlative recognition based on knowledge and context is introduced here so as to provide a rather reliable outcome . When a wire is found , an adaptive tracking along the wire detects the faults in case they exist . The following diagram shows the whole process.

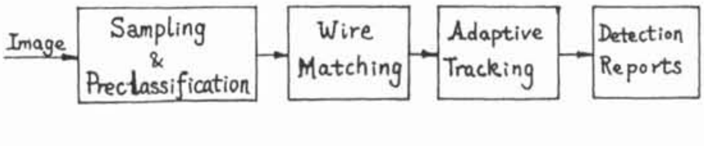

Fig 1

2 Algorithm Analysis

2.1 The Sampling and Pre-classification

It is true in actual cases that the wires in a picture usually pass through the central areas of the pictures , thus we can take some lines of image elements which go across the wires as samples to be analysed (as shown in Fig 2). The sample lines are taken from the central area of the picture . If they are near the picture edge, they may possibly miss the wires (as the line L in Fig 3). For those wires whose absolute slopes are greater than 1 ,the above mentioned sampling may possibly miss the wires (as shown in Fig 4), and this will cause a failure in the recognition or tracking to be discussed later . In this case, we only need to change the direction of sampling and take horizontal sample lines (as the dotted lines in Fig 4) , while other algorithms remain unchanged . In the following discussion , we will just take the vertical sample lines as examples to demonstrate the algorithm.

The distance between two adjacent sample lines (called the sample separation distance D) is determined rather subjectively . But it follows these principles:

1> The distance cannot be too large as for most sample lines to pass through the wires;

2> The distance should not be too small so that the intersections of the sample lines and the wires can generally be regarded independent of each other.

The number of the sample lines that need to be taken should not be too small. To garantee the reliability , we generally require $N>=5$, where N is the number of the sample lines.

The pre-classification deals only with the sample points and each sample line is analysed independently (this could be done parallelly). A grey level threshold G0 dependent on training is used. Since our interest is to find the possible "overly hot" point , only those points whose grey levels are above G0 are taken into consideration, i.e, $G(X,Y) > G0$ where $G(X,Y)$ is the grey level of the point (X,Y) . Another knowledge is also utilized here---the thickness of the wire. Based on training, we can determing an average thickness parameter W of the wires in the pictures. Actually , the thickness of the wire does not vary greatly , which shows certain stability in the value of W.

On each sample line , the algorithm searches the strings of consecutive points whose grey levels are all above G0 . THe coordinates of the first point of each string is recorded. This is called a clue point.

A string S on Li can be represented as:

$$Si=\{(Xi,Y_l)\,|\,G(Xi,Y)\geq G0,Y_{l+1}=Y_l+1,\ l=1,2,...,M,G(Xi,Y_{m+1})<G0\}$$

where M is the string length . If M W, two or more points in the string are recorded . The distance between two adjacent clue points in a string is W . If the distance between two clue points is less than W/2 , the two points are emerged into one , and the average coordinates are recorded instead . We can see from the above sampling that even if the wire passes through a large noisy area (the grey levels of many background points are above G0) , sufficient clues leading to a correct recognition have been preserved , although some background points may be mistaken for clue points. In our later discussion, we will show that these noise points can be rejected by the recognition and tracking algorithm.

2.2 Further Recognition Based on Knowledge and Context

Assuming that L1, L2, ..., Ln be the sample lines, and their X-coordinates be X1, X2, ..., Xn respectively. We choose another line L0 in the picture to be the reference line.For Li (i=1,2,...,N), we apply a knowledge-context-supporting

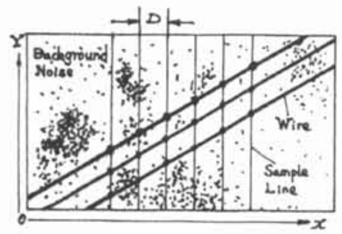

Fig 2

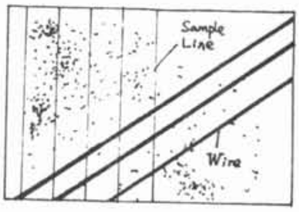

Fig 3

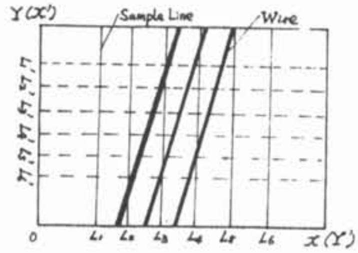

Fig 4

pattern recognition algorithm to find the wires. The knowledge utilized can be described as follows:

A> The slope K of the wire satisfies

$$|K| \leqslant 1 \qquad (2\text{-}02)$$

For those whose $|K| \geqslant 1$, a rotation of 90° in sampling direction will solve the problem, as mentioned before.

B> The points on one wire form a linear structure. If some clue points belong to one wire, they must be colinear.

We now take L1 as an example to tell how each sample line is analysed . All the clue points on L1 are seqentially analysed (they could be done parallelly). For each clue point (X1, Y1) on L1, a search for clue points is made in range R2 on L2, where

R2=[Y1-D-W,Y1+D+W] (2-03)

Note that the distance between L1 and L2 is D. Obviously , the search range has exceeded that constrained by $|K| \leqslant 1$, so in fact the algorithm based on vertical sampling can find certain wires with absolute slopes $|K|$ somewhat larger than 1.

D and W are sample separation distance and the thickness respectively.

If the search fails to find any clue points in R2 on L2 , the algorithm goes on to the next clue point on L1 and a similar search is applied . If the search finds a clue point (X2, Y2) in R2 , a line l1 passing through (X1,Y1) and (X2,Y2) can be determined immediately:

Y-Y1=K*(X-X1) (2-04)

where K=(Y2-Y1)/(X2-X1) is the slope of l1. l1 is called a clue line which might be the expected wire . The intersection point of l1 and Li (i=0,3,4,...,N) can be calculated by the following formula:

Yi=Y1+K*(Xi-X1) (i=0,3,4,...,N) (2-05)

where Xi is the X-coordinate of Li. Specially, the intersection A=(X0,Y0) of l1 and the reference line L0 is called a reference point.

The algorithm then searches in Ri to see if there are any clue points, where

Ri=[Yi-W/2,Yi+W/2] (i=3,4,...,N) (2-06)

and yi is the Y-coordinate of the intersection point (Xi,Yi) (i=3,4,...,N) . For this search, a precedence parameter P is introduced.P is defined as follows:

P=P+T (2-07)

T is a search checker which is defined as:

T=1--the search finds a clue point in Ri

T=0--the search finds no clue points in Ri

The initial value of P is zero. Obviously, the possible maximum value of P is N-2,which indicates a good matching with a possible wire.

When all the ranges Ri (i=3,4,...,N) have been searched through, the value of P is checked. If $P \geqslant M$---where M<N-2 is the tolerance limit which depends on the reqirements of the algorithm's reliability---the coordinates of the reference point A (now called a candidate) and its corresponding parameters K as well as P are recorded (The data structure of the record will be discussed later) . Here K is the average slope of the lines formed by (Xi,Yi) . It is calculated based on LMSE principle:

$$K=\frac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sum_{i=1}^{n}(X_i-\overline{X})^2}$$

where $\overline{X}$, $\overline{Y}$ are the mean values of Xi and Yi respectively. n is the number of clue points found in Ri.

Next the algorithm returns to R2 and resumes its search for more clue points. Similarly, more clues may be recorded , and so we may get (A1,K1,P1), (A2,K2,P2), (A3,K3,P3),...

When R2 has been searched through, the algorithm turns to the next clue point on L1, and applies the same search as mentioned above.When L1 is finished, similar analysis is made to L2,L3,...,Ln . Before recording the data, the algorithm first checks through the database to see if the present data is already in file. New data is added into the records . If Ai is too close to Aj , we generally regard them as identical. So is with Ki and Kj.

The data structure of the record taken is shown as follows:

As shown in Fig 5 , there two items for each candidate . One is its Y-coordinate,the other the slope.We assume that P1>P2>...>Pi.

From what has been discussed above, we know that even if some points on a wire are corrupted by noise --- their G(X,Y)s are below G0 (The possibility of most importint samples being corrupted is small in fact)---we can still find the clues of the wires according to the contextial information based on multi-sample analysis. As shown in Fig 6 , assuming that B1 and B2 are corrupted while B3, B4, B5, B6, B7 are not. In this case, the analysis of L1 and L2 will fail to find the wire. Nevertheless, the analysis of other sample lines will find the candidate A. If we let the tolerance limit P be 4, the algorithm will find the wire.

2.2 Adaptive tracking and fault-detection

The tracking strategy always chooses the clue with maximum P first. So the tracking order is P1, P2, ... Now that the candidate Ai and the slope K are known, the task is simply to track along this possible wire. The estimation of the

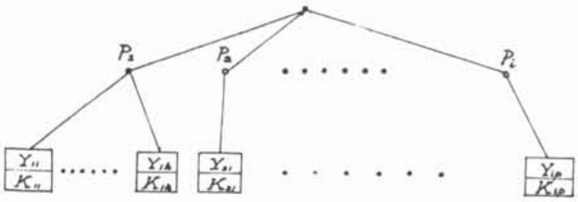

Fig 5

next expected clue point can be given by three formulas:

1> Origin Prediction

$$Y_n = K_{n-1} * (X_n - X_0) + Y_0 \qquad (2\text{-}09)$$

where X_0 and Y_0 are the coordinates of A_1. K_{n-1} is the tracking slope

$$K_{n-1} = (Y_{n-1} - Y_0)/(X_{n-1} - X_0), \quad K_0 = K_1 \qquad (2\text{-}10)$$

This formula has a rather strict restriction on the wire's linearity.

2> One-step Prediction

$$Y_n = K_{n-1} * (X_n - X_{n-1}) + Y_{n-1} \qquad (2\text{-}11)$$

Since X_n and X_{n-1} are adjacent in picture, $X_n - X_{n-1} = \pm 1$, we have

$$Y_n = \pm K_{n-1} + Y_{n-1} \qquad (2\text{-}12)$$

It is quite clear that this formula provides a very flexible tracking.

3> K-step Prediction

$$Y_n = K_k * (X_n - X_{n-k}) + Y_{n-k} \qquad (2\text{-}13)$$

(Of course, the first K-1 points are still determined by 2> or 1>).

When the prediction point (X_0, Y_0) is determined, the tracking algorithm searches for points whose $G(X,Y)$ is above G_0 in the range $[Y_p - W/2, Y_p + W/2]$. When such a point Z (clue point) is found, the tracking algorithm searches for a string of consecutive clue points the first element of which is Z. The central position of the string and the average grey level of the string elements are taken into the records.

We experimented with these three formulas. The results shows:

A> If the wires are not so regular (not very straight), the tracking based on 1> sometimes may go wrong.

B> If there is linear background noise that is quite near or connects the wires (particularly those with slopes around the value of K), the tracking based on 2> sometimes might be misled.

C> Formula 3> can avoid to certain extent the shortcomings of the former two. It keeps the quality of linearity and the flexibility or adaptability as well. Thus it gives a better result (See Fig 7).

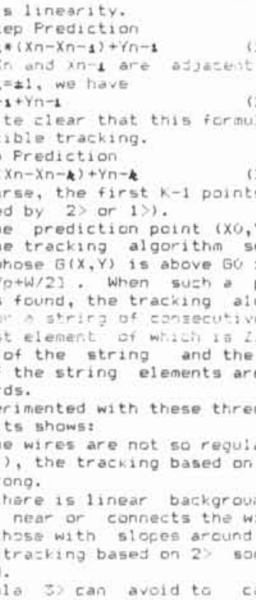

Fig 6

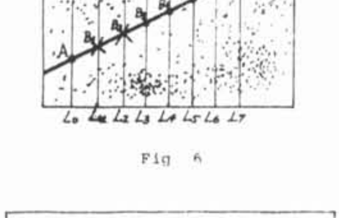

Fig 7

The "Abnormal Case Dealing" deals with the case of track missing. If no clue points are found, the algorithm goes on to search accor- to the formula , just as if the prediction point were the clue point. But it starts to count the failure times. When the consecutive failure times exceeds certain bound, or if there are too many such cases as track missing , the algorithm will deny the present tracking and turns to the next one.

When a wire is tracked over, the average grey level E and the corresponding mean-square-error σ are calculated. If

$$G(X,Y) - E \geq C * \sigma \qquad (2\text{-}14)$$

where $G(X,Y)$ is the grey level of the recorded point , and C is the confidence coefficient dependent on specific reqirements , the present point is regarded as a fault point and corresponding data is printed out.

Since the high-tension wires are parallel, their slopes are generally the same or very close to each other . Thus, when one wire is inspected over , the information implied in its slope K may help reduce the search for a next wire . We use the knowledge to increase (or even maximize) the precedence of those clue points whose K are identical or close to K . When two of the wires are inspected , we can almost confirm that K is the right slope of the expected wires . So more measures can be taken to reduce the search if necessory (e.g. when K is confirmed, the algorithm next only searches those with the slope K . This will surely reduce the search greatly).

If no wires are found , the algorithm will change the sampling direction and adopt horizontal sample lines . Similarly, a recognition and tracking is applied.

3 Discussion and Conclusion

From the above discussion , we can see that the algorithm based on multi-sample analysis has the following features:

1> Since sample analysis is introduced, computations involved have been reduced greatly , and a quick matching is obtained. In actual experiment, for a 512 512 image , averagely speaking , a matching is finished within a few seconds (realised on IBM-PC/XT).

2> With the help of knowledge and contextial information , the algorithm can successfully reject quite many noise points and detects the wires out of the background . In actual experiments , it shows impressive reliability which is very important in practical application.

3> The algorithm does not consume much of the computer s memory (only certain important coordinates , etc, are stored). Thus , it can easily be realised on micro-computers.

4> The recognition can be made much more quickly if parallel operation is introduced , for a large part of the computations could be done parallelly.

Reference

[1] J.CANNY, Finding Edges And Lines in Images, M.I.T Technical Report No.720, 1983.
[2] R.DUDE, P.HART, Pattern Recognition And Scene Analysis, Wiley, NewYork 1973.
[3] M.HUECKEL,An Operator Which Locates Edges in Digitized Pictures, J.A.C.M, Vol.18. pp113-125, 1971
[4] J.SHEN; S.CASTAN, Fast Binary Correlation Algorithm for Stereo Vision, 5th Scan Conf. on Image Analysis, Stockholm 1987.
[5] Rosenfeld,A. and Kak, A.C.(1976), Digital Picture Processing, Academic Press , New York.
[6] Devijver, P.A. and Kittler, J. (1982),Pattern Recognition:a statistical approach, Prentice Hall, London.
[7] Chiffineni, C.B (1981). On the extraction of features from slowly wandering patterns. IEEE Trans. Aerospace and Electr.System ,AES-17, 461-466
No.720, 1983.
[2] R.DUDE, P.HART, Pattern Recognition And Scene Analysis, Wiley, NewYork 1973.
[3] M.HUECKEL,An Operator Which Locates Edges in Digitized Pictures, J.A.C.M, Vol.18. pp113-125, 1971
[4] J.SHEN; S.CASTAN, Fast Binary Correlation Algorithm for Stereo Vision, 5th Scan Conf. on Image Analysis, Stockholm 1987.
[5] Rosenfeld,A. and Kak, A.C.(1976), Digital Picture Processing, Academic Press , New York.
[6] Devijver, P.A. and Kittler, J. (1982),Pattern Recognition:a statistical approach, Prentice Hall, London.
[7] Chiffineni, C.B (1981). On the extraction of features from slowly wandering patterns. IEEE Trans. Aerospace and Electr.System ,AES-17, 461-466
[8] D.MARR,E.HILDRETH, Theory of Edge Detection, Proc R. Soc. London, Vol, B207, pp187-217, 1980.
[9] W.PRATT,Digital image processing, New York, 1978.
[10] K.S.Fu, Syntactic Pattern Recognition And Applications, Prentice Hall, 1981.
[11] K.S.Fu, Syntactic Methods in Pattern Recognition, Academic Press, 1974.
[12] A.Rosenfeld, Pictures Languages, Academic Press, 1979.