Chemical Literature Data Extraction: The CLiDE Project

P. Ibison, M. Jacquot, F. Kam, A. G. Neville, R. W. Simpson, C. Tonnelier, T. Venczel, and A. P. Johnson*

School of Chemistry, The University of Leeds, Leeds, LS2 9JT United Kingdom

Received July 28, 1992

Chemical information, especially that concerning chemical reactions, is becoming increasingly available in a variety of computer-readable databases. However, the creation of these databases is a time-consuming and expensive process. CLiDE (Chemical Literature Data Extraction) is a new software project to help solve the problem of building substance and reaction databases. CLiDE uses a combination of imaging and artificial intelligence techniques to recognize a range of chemical diagrams and extract the information they contain. The steps necessary to transform a chemical structure drawing into a computer-readable output are detailed. Several examples are given to illustrate the scope of the current work.

GENERAL DESCRIPTION

On-line and in-house chemical databases provide the modern chemist with powerful tools to access the literature. These tools allow the chemist to make both rapid and complex searches for the information required. Information typically stored in chemical databases might include molecular structures, chemical reactions,[1] or reaction schemes. Textual information such as an abstract, references, or keywords usually accompanies the purely structural information.

Abstracting information from the literature and manually entering it into these databases is a laborious and expensive task requiring a trained chemist. CLiDE (Chemical Literature Data Extraction) is a new software system under development that attempts to solve this problem by computerizing the procedure. Programs of this type have been developed extensively in other subject areas.[2,3] Within chemistry, McDaniel and Balmuth[4] have described a commercial product for the interpretation of scanned chemical structures [Kekulé: Optical Chemical (Structure) Recognition]. Contreras et al.[5] have presented details of a similar program that has been applied to simple molecular images already isolated from a journal page.

The aim of CLiDE is to process whole page(s) of chemical information from journals and books in order to extract the chemical structures, reaction schemes, and other relevant chemical information. Another important requirement is to perform this task as far as possible without human intervention. In those cases where some ambiguity exists, the program is designed to prompt the user for additional information.

The different features which might be found on a page, such as structures, reaction schemes, diagrams, and tables, cannot all be processed in the same way. Their characteristics are so different that a specific process is required for each of these object families. However, some processes will be able to use subprocesses already implemented for dealing with simpler objects. The interpretation of a single chemical structure is a fundamental process that is also required for interpreting reaction schemes and for the understanding of reaction tables containing chemical structure diagrams.

The interpretation of single structures is, at present, the main achievement of the CLiDE project and opens the way for the processing of more complex objects. The CLiDE program is able to produce a computer-readable file after scanning most organic chemical structures. As will be shown, this includes complicated structures involving crossing bonds, molecular groups, and generic atoms.

The tasks performed to generate the connection table corresponding to a drawing of a chemical structure are detailed below. Both isolated structures and structures embedded in text or graphics can be processed with the current program.

SYSTEM OVERVIEW

The CLiDE program establishes the connection table of a scanned chemical structure in three steps: a recognition phase, a text grouping phase, and an interpretation phase. The general process is shown in Figure 1, and the route from a scan of an initial structure to the final connection table is shown in Figure 2. The details of the individual steps shown in Figure 2 are discussed in the following sections.

During the first phase, individual features like characters, lines, curves, etc. are recognized. These features are called primitives and are divided into characters and graphic primitives. The characters are interpreted and the complex graphics broken into components. The next phase, text grouping, groups the characters into words, the words into lines, and finally lines into blocks of text. The final phase of structure recognition, the interpretation phase, converts the primitives into higher level data types, termed items (atoms, bonds, wedge bonds, etc.). The groups of items are assembled into our internal representation of the information, i.e., the connection table of the structure. The extracted information is finally displayed graphically for verification by the user and written into a disk file.

CLiDE is written in C++ and currently implemented on a SUN SPARC work station. The images are scanned by an Agfa Focus S 800GS scanner at 300 dpi (dots per inch) resolution.

PRIMITIVE RECOGNITION

The binary scan of a journal page produces a fine grid of black or white dots, called pixels; the grid itself is referred to as a bitmap. Recognition of primitives in the bitmap is the most time-consuming task of the CLiDE process due to the amount of data to process and the local view the computer has of the data. Using a scanner resolution of 300 dpi produces approximately 1 MB of data to be processed for a typical journal page.

During the first step of primitive recognition, the bitmap image is segmented into connected black regions, which we term connected components. The connected components are represented by their interpixel contours, which are defined as the discontinuous lines surrounding the black regions at the

0095-2338/93/1633-0338$04.00/0

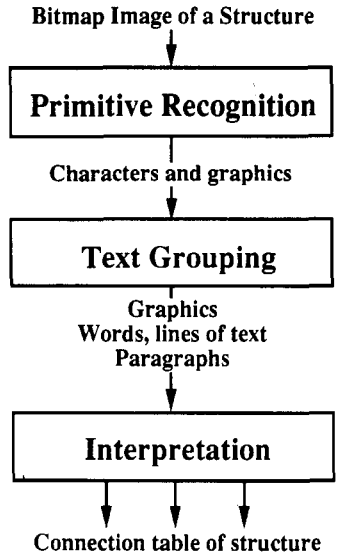
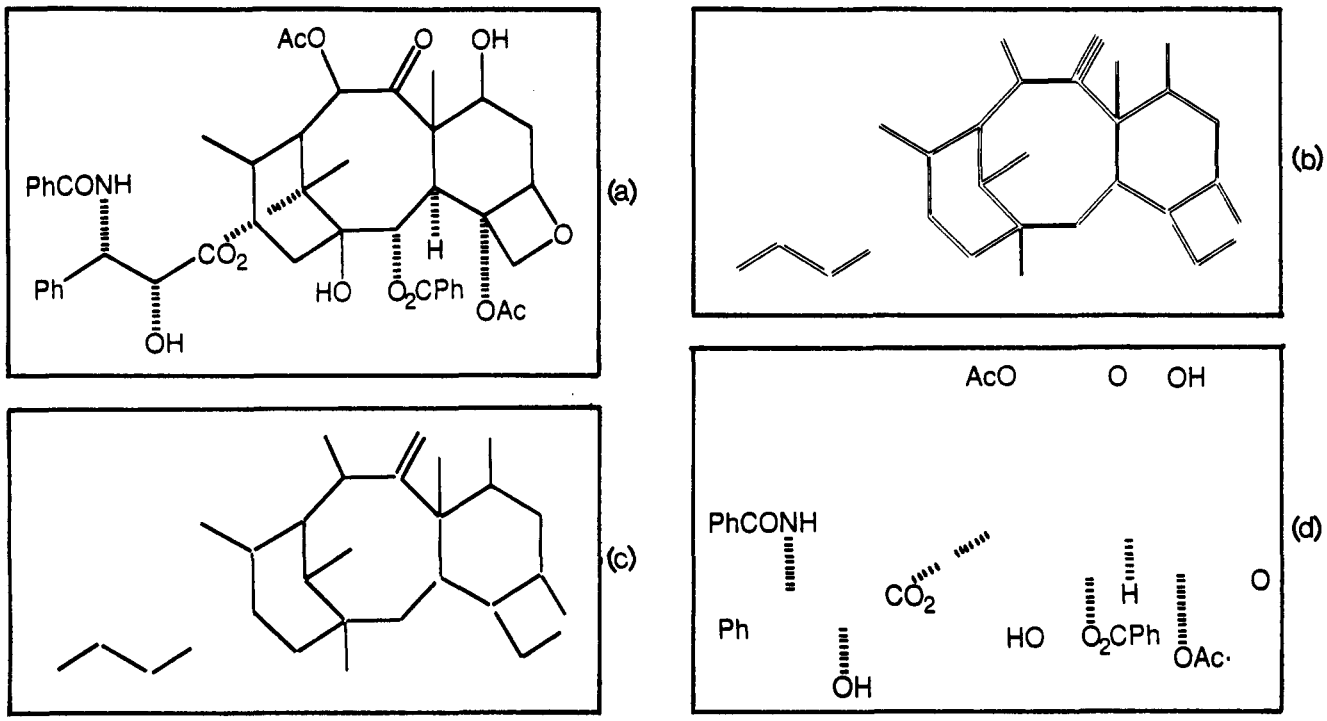


Figure 1. CLiDE system for chemical structure interpretation proceeds in three phases. These phases combine to transform the original bitmap image of a structure into a chemically useful format.

edges of the outermost black pixels. This representation significantly reduces the size of memory needed to store an image. The contours are defined as the coordinates of a starting point and a sequence of four directions (N, S, E, W). The method of Ahronovitz–Bertier–Habib[6] is used to create the contour code and to find the connected components; this is shown in Figure 3. Each connected component may be further divided into graphic primitives. The recognition of graphic primitives can be done on the connected components separately because each primitive is, or belongs to, only one connected component. The primitive recognition is performed using the contours without the reconstruction of the original bitmap. Connected components are divided into subclasses, the most important being characters, graphics, and dashes. Due to the embedding of isolated characters in graphic regions, text/graphic separation methods based on string detection[7] could not be used. The distribution of connected component sizes in the image is analyzed, and the size of the largest character is estimated. The separation of connected components into the three classes is then done using this size and



Entry	Atom	H's	No. Bonds	Bonded To
1	6	1	3	[1,0,2][1,0,8][1,3,9]
2	6	0	3	[1,0,1][1,0,3][1,0,10]
3	6	0	4	[1,0,2][1,0,4][1,0,11][1,3,12]
4	6	1	3	[1,0,3][1,0,5][1,2,13]
5	6	1	3	[1,0,4][1,0,6][1,2,14]
6	6	1	3	[1,0,5][1,0,7][1,3,15]
7	6	0	4	[1,0,6][1,0,8][1,3,16][1,2,17]
8	6	1	3	[1,0,7][1,0,1][1,3,18]
9	408	0	1	[1,0,1]
10	8	0	1	[2,0,2]
...	...	...	...	...

(e)

Figure 2. (a) Original scanned image (redrawn from *J. Org. Chem.* **1990**, *55*, 3–5). (b) Straight contour fractions. (c) Straight line primitives. (d) Dashed lines and characters. (e) Representation of part of the structure as a connection table.

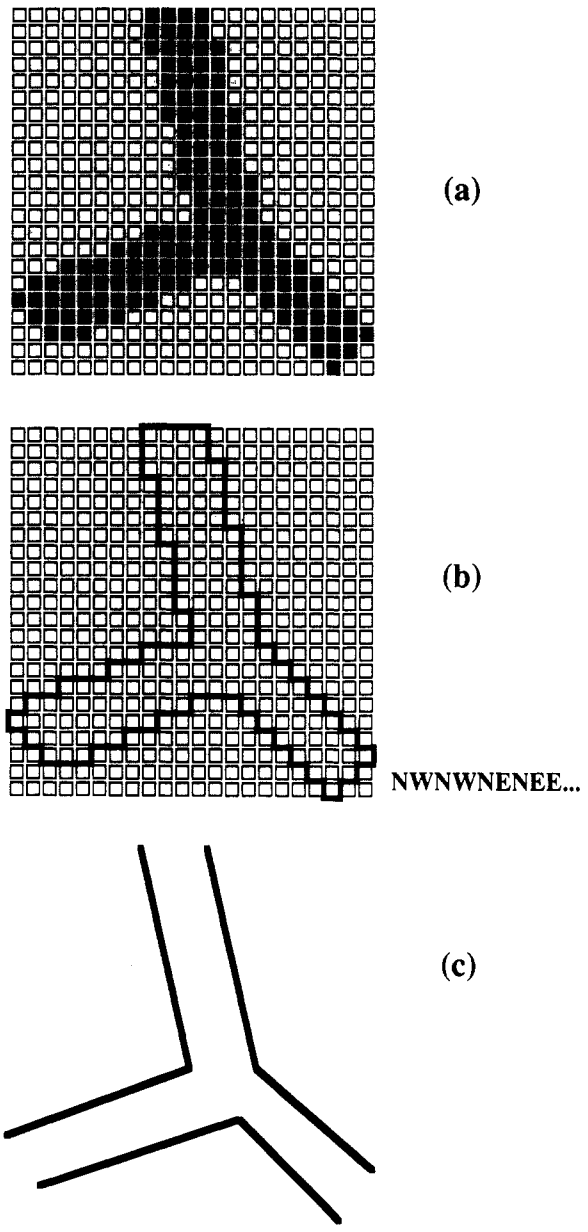


Figure 3. (a) Bitmap containing one connected component. (b) Contour of the connected component seen in panel a. (c) Straight fractions of the contour seen in panel b.

information about the relative height, width, and contour length of each connected component. This ensures that the separation is independent of the image size. This method results in a reasonably accurate separation. Any errors are corrected during the recognition step by moving those connected components which cannot be interpreted from one group into another.

Two methods described below were developed to find the graphic primitives in the connected component classes.

The first method takes graphic connected components and extracts the elementary lines and curves (graphic primitives). This process of decomposing line drawing images into primitive graphic elements such as lines and curves is often called image segmentation. Most methods for image segmentation start by reducing the width of the line-like objects from many pixels to just a single pixel. This process is called thinning or skeletonization.[8,9] This skeleton of the image is then segmented into straight lines and curves by finding the dominant points. We did not follow this method because it is unsuitable for chemical diagrams: Thinning algorithms are time-consuming, and during the process important information is lost about

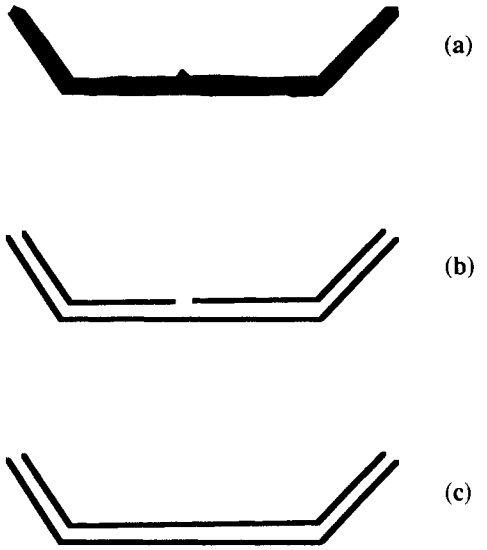


Figure 4. (a) Connected component having three straight lines, with noise occurring in the center of the middle line (taken from a scanned image). (b) Straight fractions for panel a. Note that there are two fractions for the top border of the horizontal line. (c) Straight fractions of panel a after joining the two fractions belonging to the same lower border of the horizontal line.

the diagram, e.g., wedged and wavy bonds are important in chemical structures and this information is not present in the skeleton of a drawing.

In place of the skeleton we use the contours of the image. These are cut into straight and curved fractions (Figure 3c). The creation of these contour fractions is an important part of the graphic primitive recognition process. For each contour, a polygon is created in such a way that each point of the original contour is within a certain distance of a side of the polygon. If this threshold value is well chosen, straight parts of the contour result in long polygon sides, while curved parts are approximated by consecutive short sides. A method similar to that of Sklansky and Gonzalez[10,11] is used to create the approximating polygon. Long polygon sides are selected as straight contour fractions, and consecutive short sides are merged into curved fractions. Individual short sides are not used for primitive detection. In an ideal case there are two fractions created for each line-like primitive, which are the two borders. Fractions belonging to the same primitive are found by searching pairs that are adjacent. During this step, the errors of the fraction creation phase are corrected. If the image is noisy, the fraction detection phase can create more than two fractions for one primitive by incorrectly cutting one border at an internal point (Figure 4). In such a case, both parts will be side-by-side with the other border, and an attempt can be made to join them. Alternatively, if the two parts cannot be joined, it indicates that they belong to different primitives sharing the fraction on opposite sides, and the shared fraction is cut into two according to the primitives (Figure 5). Finally, each primitive is described by the two borders. The coordinates of the line end points, the line width, and the line shape are determined from these two borders.

The second method is the processing of dash-like connected components to find the chemical dashed bonds. Dashed line detection is performed on the dashes group which contains all small connected components. Most of them are dashes from dashed lines, but some are small characters such as ",", ".", and "-". The Hough transform method[12,13] is used to select the dashes situated along a line and to determine the parameters of this line. A single coordinate is used to represent the position of each 'dash', and all dashed lines containing at

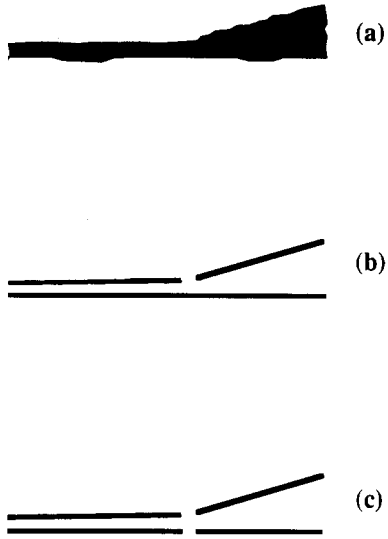


Figure 5. (a) Connected component of a straight line and a wedge line (taken from a scanned image). (b) Straight fractions for panel a. Note that there is only one fraction for the bottom border of the line and the wedge. (c) Straight fractions of panel a after cutting the one belonging to both primitives.

least three dashes lying close together are found. The number of dashed lines does not need to be known in advance. The process of finding the resulting dashed lines of the image is shown in Figure 6. The connected components remaining in the dashes group are now treated as small characters and are subsequently moved into the characters group.

This analysis, as implemented, recognizes graphic primitives such as straight lines, dashed lines, triangles, wedges, and to a limited extent, curved lines.

The remaining connected component class containing characters is analyzed by a separate process; Optical Character Recognition (OCR).[14] We have developed capability specific to our requirements: a neural network has been trained to recognize alphanumeric characters, represented by density matrices derived from the character bitmaps. Characters are located automatically in the bitmap and are interpreted using the network. Any unassigned or misassigned characters are corrected manually.

The accuracy of the automatic character recognition and the graphic primitive recognition is difficult to estimate as this is dependent on the test images used. Currently for a chemical structure taken from the literature, we recognize approximately 90% of the distinct characters correctly. For graphic primitive recognition, all lines longer than threshold length are correctly recognized. For dashed lines, we require that they contain at least three dashes. The recognition of thin wedged lines is sometimes difficult.

At this stage of the CLiDE process, all of the elementary (chemical) graphics and characters have been identified. The next phase assembles these primitives into chemically meaningful groups and finally complete structures.

GROUPING PHASE

The grouping of characters creates small text regions, such as paragraphs, chemical strings, and structure labels. The text grouping begins by grouping into words those characters which are adjacent and on a line. Words are then grouped into lines, and finally lines are grouped into blocks of text. Each character is associated with one block of text. For each of these grouping phases, parameters are necessary, e.g., the maximum separation of characters in a word. These parameters need to be set correctly, within certain ranges, for this

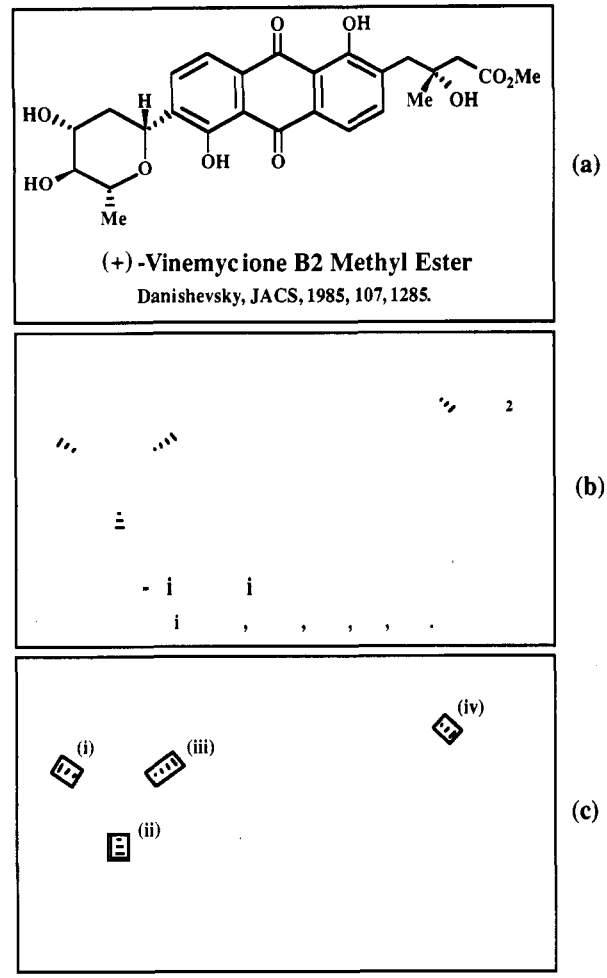


Figure 6. (a) Original image of a chemical structure with some associated text. (b) Connected components in the dash group before dashed line detection. (c) Connected components in the dash group after dashed line detection. Each of the four groups of dashes, (i)–(iv), now corresponds to a single dashed, wedged line.

process to work successfully. For example, if the maximum allowed separation of characters in a word is too small, then each letter will be classed as a word; while if the maximum is too large, each line of text will be classed as a word. An optimum set of parameters for this grouping process is used for each type of document, with the user being given the opportunity to alter them when necessary.

Once the text blocks are identifed, we are particularly interested in those which consist of only one word, because they may be text in, or associated with, structures. Those blocks which are clearly paragraphs of text are not used by the program at present.

INTERPRETATION PHASE

A chemical structure is composed of two types of objects: graphics and text. As has been discussed previously, we have processed the information on the page to produce text items and graphic primitives that may form part of a chemical structure. It is the function of the interpretation phase to correctly identify the chemical context of these items and primitives and to build a connection table from them. The text occurring in a structure may correspond to an individual atom or a molecular group (e.g., Cl or Me, respectively) and is referred to as a superatom. All of the relevant information concerning superatoms is contained in a database. The graphic primitives are, or are part of, chemical bonds in a structure.

It is relatively straightforward to interpret these graphic

Table I. Superatom Information for Several Superatoms Stored in the Database

superatom	O^-	O	HO	OMe	COCO
code	8	498	426	462	499
connections	2	1	1	1	2
equivalent	O	O	OH	OMe	COCO
hydrogens	0	0	1	0	0
charge	−1	0	0	0	0
stereochemistry	0	0	0	0	0
negative charges allowed	0	1	0	0	0
positive charges allowed	0	1	0	0	0
letter bonded list	1	1	2	1	1, 3
connection table				8 [1,0,2]*	6 [1,0,3] [2,0,2]*
				6 [1,0,1]	8 [2,0,1]
					6 [1,0,1] [2,0,4]*
					8 [2,0,3]

primitives as chemical bonds. The main complications arise where lines are adjacent (double or triple bonds) or are broken by the presence of a crossing line. The resolution of these problems will be discussed later.

Thus, the two main tasks during the interpretation phase are the identification of the text present in the structure and (using this information and the list of chemical bonds) the construction of the complete connection table.

Some interpretation is left until after the chemical structure connection table has been built. In particular, text strings near a structure can be interpreted in context. A "+" or "−" sign near a vertex will signify a charge, a text string may be a name, and a numeric label may be a structure number. One particular text block, the generic block, is also identified during this phase, and syntactic rules are used to interpret the chemical information in this block. Generally, each line of a generic block represents an individual structure and contains one or more substitutions (e.g., R^1 = Me, R^2 = OEt). For each structure a substitution of the superatom(s) for the generic string(s) is performed, i.e., Me for R^1 and OEt for R^2 in the above example.

(a) Superatom Identification. The identification of superatoms in the CLiDE process is done by referring to a look-up table. In fact the task is more than identification; the database not only allows one to check the validity of the superatom items but also contains information required to build the connection table. Although such a database can never be complete, it allows the vast majority of chemical structures to be interpreted correctly and efficiently. Each superatom is represented by a card (Table I) containing several items of information listed as follows:

(1) A code number, which is the atomic number for elements and a number greater than 400 for other superatoms (uniquely assigned when added to the database).

(2) The number of external bonds required by the superatom (valency for the atoms).

(3) The description of the charge and stereochemistry of the superatom.

(4) The number of negative and positive charges allowed. This information will be used for further checking of the validity of the connection table.

(5) The number of explicit hydrogens bonded to the superatom.

(6) An equivalent item which will replace the current superatom in the reformated connection tables in order to obtain a consistent representation. For example, "C", together with the number of hydrogens and the charge, is equivalent to "CH", "CH_2", "CH_3", "C^-"; "C" is the equivalent item of these superatoms.

(7) The number(s) of the letter(s) of the item that are used to create the correct bond–text connections. For example, "HO" is externally bonded by the second letter, so the number two is stored while the carboxyl group "CO_2" is externally bonded twice, once by the first letter and once by the second, so both numbers are stored.

(8) Connection table. The connection table for a superatom is encoded as follows: For each atom in the superatom, the connection table specifies its atomic number, and for each bond the atom makes there is a code. This code is three numbers indicating the bond order, the bond style (normal = 0, wedged, etc.), and the number of the atom that the bond is made to. A flag (*) is used to indicate if an external connection is made from this atom.

When required, the connection table of the superatom will be used to produce the expanded connection table for the structure, i.e., the connection table in which all the non-hydrogen atoms are explicitly described. The totally expanded connection table contains only superatoms which have no subconnection table, i.e., which are in fact atoms.

When a superatom is not found in the database, CLiDE prompts the user to enter the required information, which is then added to the database. Currently the database contains approximately 200 superatoms. The knowledge provided by the database helps solve ambiguities during the construction of the connection table. For example, when more than one bond is close to a superatom, knowing which letter in the item has to be externally bonded is helpful. Further reformatting would be difficult without interpretation of the superatoms, e.g., the same superatom is often referred to by different names in chemical structures. In order to obtain a consistent representation, it is necessary to give a unique name for each superatom. Finding the equivalent name for each superatom is done by referring to the superatom database.

(b) Building the Connection Table. The internal representation of the information concerning chemical structures identified by CLiDE is a connection table which describes each atom and its environment. The connection table, once reformatted, can be stored for searching and external use.[15]

The information defining each bond (order, style, coordinates) and the superatom information are combined to form the connection table. The bond–atom connections are determined before the bond–bond connections.

Other studies[5] base the recognition of a join between two bonds or between a bond and an atom on the distance separating the components; gaps of less than a certain threshold result in the components being joined. Our early experiments showed that this is not accurate enough for reliable automatic operation. For example, if the primitives are not recognized correctly, there may be larger gaps between primitives than are actually present in the structure. Structures are also sometimes drawn in an ambiguous way, e.g., two bonds may closely approach a monovalent superatom. Consider the example in Figure 7: The bonds (i), (ii), and (iii) are all close to the superatom OAc, but bonds (ii) and (iii) are not pointing toward the bonding atom (oxygen) of OAc. This illustrates that the direction and the distance of the bond in relation to the superatom should be considered.

The superatom database is consulted to determine the free valence (n) and the coordinates of the bonding letter(s) of the superatoms in the structure. This gives the number of bonds

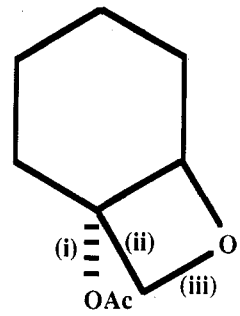


Figure 7. Example of a structure in which the superatom is closer to two (solid) bonds (ii and iii) than to the dashed bond (i) to which it is actually joined.

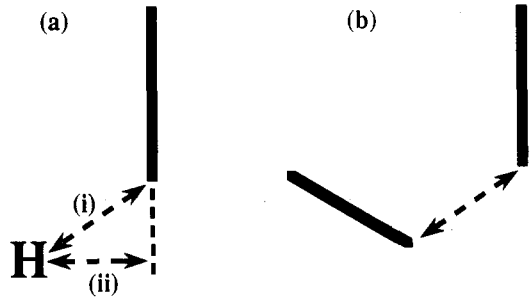
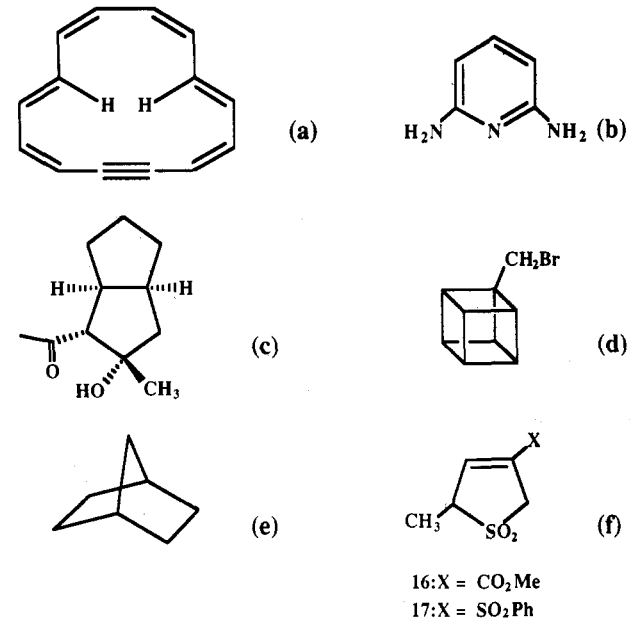


Figure 8. (a) Separation (i) and perpendicular distance (ii) of an atom from a bond end are used to determine if they should be connected. (b) Bond ends are joined together if they are sufficiently close to each other.

that should be joined to each superatom and guarantees that the correct bonds are chosen. In the second step, '*m*' closest bonds are chosen where $m \geq n$, which means that the number of candidate bonds being considered may be more than the number that the superatom requires. Not all the bond ends in the structure will be considered, because some of them are obviously too far from the superatom. For each candidate, the perpendicular distance from the superatom is calculated. A small perpendicular distance between the bond and a superatom means that the bond is pointing toward the superatom (see Figure 8a). In order to pick out *n* bonds from *m*, a scoring function is used to balance the two factors, separation and direction of the bond from the superatom. The joining of two bonds is more straightforward; bond ends which are close are joined together (see Figure 8b).

After precipitation, the information about a structure is stored as a connection table. Many types of bonds are perceived including single, double, triple, thick, wedged, dotted, dashed, dashed wedged, and wavy. Each bond type is represented by a number, e.g., 1 for single bonds, 2 for double bonds, etc. For most of these bonds, the sense of the direction of the bonds is not important, but for the wedged bonds and dashed wedged bonds, the connection table can distinguish which atom is situated at the apex of the wedge and which atom is situated at the opposite end. Other chemical characteristics, e.g., stereochemistry and charge, are also stored in the connection table. The 2D coordinates of the structures obtained from the original scan are retained so that the structure may be redrawn. Finally, the remaining small pieces of text are examined. If any text items are close to the completed structure, the information is retained with the connection table as potential structure labels or information.

The last step of the global process consists of converting our internal information representation (i.e., connection tables) into a suitable format which will be the correct input format for the structure databases. The current version of CLiDE outputs chemical structure information in MOLfile,[16] Chem-



16:X = CO_2Me
17:X = SO_2Ph

Figure 9. Set of structures which CLiDE is currently able to process successfully.

Draw,[17] and CLiDE internal formats and can create a Postscript file of the redrawn structure.

Most of the chemical reaction databases such as ORAC[1] or substance databases allow substructural searches. Such inquires can only be performed with full success on expanded structures or on structures containing superatoms known by the database management system. The program is able to convert superatom synonyms into standard forms (e.g., MeO and OCH_3 are replaced by the unique superatom OMe). It can also create partially or totally expanded connection tables (i.e., connection tables in which all the atoms and bonds are detailed). To perform these tasks, CLiDE is connected to the extendable superatom database described above, which stores the necessary information about the superatoms currently encountered in the literature.

EXAMPLES

An example set of structures which have been successfully processed using CLiDE is presented in Figure 9. These structures have been chosen to illustrate the scope of the current program. Figure 9a is a simple structure including single, double, and triple bonds. The only text is the two hydrogen atom labels. In the second structure, Figure 9b, the two text strings NH_2 and H_2N are determined to be equivalent using information contained in the superatom database. The stereochemical information contained in dashed and solid wedged bonds shown in Figure 9c is correctly identified. Parts d and e of Figure 9 present two chemical structures containing bonds that are crossed. The first example is relatively straightforward since one bond 'cuts' the other. The second example is chemically ambiguous since the crossing point could be misinterpreted as being a carbon atom. We have developed a set of rules to correctly interpret both of these cases. These rules include the proximity, length, collinearity, and ring membership of potential crossing bonds. The last example presented, Figure 9f, is a structure including generic text. The generic text is parsed, and the appropriate replacement groups are identified and stored. Substitutions of the two superatom groups for X are made into the structure, resulting in two connection tables. The current version of CLiDE can successfully parse most of the common generic text blocks.

Also in this example is a superatom (SO_2) which requires two connections to the structure through the sulfur atom. This is also correctly performed.

CONCLUSION

CLiDE is able to produce a connection table from a scanned image of a chemical structure embedded in a page of text. The CLiDE prototype has been widely tested, and our algorithms have been shown to work in a number of difficult cases. Most organic structures can be processed by the current program. A wide range of bond types are detected; solid bonds (e.g., single, double, wedged, etc.) are found by examining the contours of the graphic components, and dashed bonds are detected using a modified Hough transform. A neural network-based optical character recognition module is used to identify the characters contained within the scanned area. Grouping routines identify those characters that may belong to a chemical structure. A connection table is constructed from these pieces of text and the chemical bonds. The text items are checked against a database to facilitate this process.

Future publications will detail the identification of multiple structures on a page and the rules to handle a wide range of bond crossing situations.

REFERENCES AND NOTES

(1) Borkent, J. H.; Oukes, F.; Noordik, J. H. Chemical Searching Compared in REACCS, SYNLIB, and ORAC. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 145–50.

(2) Fahn, C. S.; Wang, J. F.; Lee, J. Y. A Topology-Based Component Extractor for Understanding Electronic Circuit Diagrams. *Comput. Vision, Graphics Image Process.* **1988**, *44*, 119–38.

(3) Fukada, Y. A Primary Algorithm for the Understanding of Logic Current Diagrams. *Pattern Recognit.* **1984**, *17*, 125–34.

(4) McDaniel, J. R.; Balmuth, J. R. Kekulé: OCR—Optical Chemical (Structure) Recognition (preprint).

(5) Contreras, M. L.; Allendes, G.; Thomas-Alvarez, L.; Rosas, R. Computational Perception and Recognition of Digitized Molecular Structures. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 302–7.

(6) Ahronovitz, E.; Bertier, M.; Habib, M. Contour Coding for Image Manipulation and Compression. *IAPR 86 IEEE 742* **1986**, *2*, 1033–5.

(7) Fletcher, L. A.; Kasturi, R. A. Robust Method for Text String Separation from Mixed Text/Graphics Images. *IEEE Trans. Pattern Anal. Mach. Intell.* **1988**, *10*, 6.

(8) Naccache, N. J.; Shinghal, R. An Investigation into the Skeletonization Approach of Hilditch. *IEEE Trans. Syst., Man, Cyber.* **1984**, *3*, 409–18.

(9) Smith, R. W. Computer Processing of Line Images: A Survey. *Pattern Recognit.* **1987**, *1*, 7–15.

(10) Sklansky, J.; Gonzalez, V. Fast Polygonal Approximation of Digitized Curves. *Pattern Recognit.* **1980**, *12*, 327–31.

(11) Venczel, T.; Jacquot, M.; Johnson, A. P. An Algorithm for Straight Line Recognition. Manuscript in preparation.

(12) Duda, R. O.; Hart, P. E. Use of the Hough Transform to Detect Lines and Curves in Pictures. *Graphics Image Process.* **1972**, *1*.

(13) Illingworth, J.; Kitter, J. The Adaptive Hough Transform. *Comput. Vision, Graphics Image* Process. **1988**, *44*, 87–116.

(14) Govindan, V. K.; Shivaprasad, A. P. Character Recognition—A Review. *Pattern Recogn.* **1990**, *23*, 671–83.

(15) Ash, J. E. Connection Tables and Their Role in a System. In *Chemical Information Systems*; Ash, J. E., Hyde, E., Eds.; Ellis Horwood Publishers: Chichester, 1975; pp 157–76.

(16) Dalby, A.; Nourse, J. G.; Hounshell, W. D.; Gushurst, A. K. I.; Grier, D. L.; Leland, B. A.; Laufer, J. Description of Several Chemical File Formats Used by Computer Programs Developed at Molecular Design Limited. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 244–55.

(17) ChemDraw, Cambridge Scientific Computing Inc.