



UNIVERSITÀ DEGLI STUDI DI BARI ALDO MORO

DIPARTIMENTO DI INFORMATICA

CORSO DI LAUREA IN INFORMATICA E TECNOLOGIE PER LA
PRODUZIONE DEL SOFTWARE

TESI DI LAUREA
IN
SISTEMI MULTIMEDIALI

**TESTGRAPHIA, SISTEMA BASATO SULL'ANALISI
DOCUMENTALE PER LA DIAGNOSI DELLA
DISGRAFIA**

Relatore:

Prof. Giovanni DIMAURO

Laureando:
Davide DI PIERRO

Anno accademico 2018/2019

INDICE

Sommario

Abstract.....	5
I. Introduzione.....	6
I.I Disgrafia.....	6
I.II Diffusione sul territorio.....	7
I.III Metodi di valutazione.....	9
II. Parametri di scrittura.....	11
II.I Grandezza della scrittura.....	12
II.II Margine sinistro non allineato.....	14
II.III Andamento altalenante della linea di scrittura.....	15
II.IV Spazio insufficiente tra le parole.....	16
II.V Angoli acuti.....	17
II.VI Collegamenti interrotti fra le lettere.....	18
II.VII Collisione fra le lettere.....	19
II.VIII Grandezza irregolare delle lettere.....	20
II.IX Misura incoerente fra lettere con e senza estensione.....	22
II.X Lettere atipiche.....	23
II.XI Forme ambigue delle lettere.....	24
II.XII Lettere ritoccate.....	25
II.XIII Traccia instabile.....	26
III. Descrizione del sistema.....	27
III.I Interazione.....	28
III.II Personalizzazioni.....	34
III.II.I Impostazioni per il parametro 3.....	34
III.II.II Impostazioni per il parametro 9.....	35
III.II.III Rimuovere il footer.....	36
III.II.IV Cambiare testo.....	37
III.III Algoritmi.....	39
III.III.I Segmentazione del testo.....	40
III.III.II Algoritmo grandezza della scrittura.....	50
III.III.III Algoritmo margine sinistro non allineato.....	51

III.III.IV Algoritmo andamento altalenante della linea di scrittura.....	52
III.III.V Algoritmo spazio insufficiente fra le parole.....	56
III.III.VI Algoritmo grandezza irregolare delle lettere.....	58
III.III.VII Algoritmo misura incoerente fra lettere con e senza estensione.....	59
IV. Risultati.....	60
IV.I Progettazione test.....	60
IV.II Validazione.....	61
IV.III Validazione dello specialista.....	63
V. Ringraziamenti.....	64
VI. Sviluppi futuri.....	64
VII. Conclusioni.....	66
VIII. Riferimenti.....	67

Abstract

La disgrafia è un disturbo legato alla scrittura che influenza dimensione e distanza dei caratteri e l'ortografia. I pazienti con disgrafia probabilmente incontrano difficoltà nelle attività motorie e riescono con criticità a riportare per iscritto ciò che pensano. I sintomi non sono rari ma spesso sono temporanei. Non esistono specifiche cause che portano alla disgrafia; essa è perciò impossibile da prevenire. Per questi motivi, diventa fondamentale diagnosticarla. I metodi tradizionali sono basati su moduli cartacei e righelli.

In questo caso di studio presentiamo TestGraphia, un sistema software utile a supportare i medici nella formulazione di una diagnosi. La visita medica può essere completata in breve tempo. In questo modo permette anche attività di screening di grandi dimensioni riducendo ogni sforzo. Per la sua facilità d'uso non è escluso che questo software possa essere utilizzato a casa, rivelando eventualmente i primi sintomi. Questo sistema permette anche un rapporto remoto tra esperto e paziente; esso è in fase di sviluppo come applicazione web.

I.I Disgrafia

Nonostante un generale spostamento verso un mondo senza carta, ci sono ancora molti casi in cui le comunicazioni scritte a mano sono frequenti. Che si tratti di penna tradizionale su carta o stilo su tavoletta, la scrittura a mano spesso è più conveniente, efficiente ed economica. Inoltre, molti documenti come fatture, tasse e questionari hanno spesso campi vuoti che devono essere digitalizzati. A differenza del riconoscimento di scrittura online che registra sequenze di tratti contestuali, il riconoscimento della scrittura offline non dispone di contesto temporale[1].

La scrittura a mano è un importante mezzo di comunicazione che consente l'espressione, la memorizzazione e la trasmissione di idee da parte degli studenti per tutto l'arco della loro carriera educativa. I bambini di scuola elementare tipicamente occupano fino al 50% della giornata scolastica in attività legate alla scrittura, alcune delle quali svolte in tempi ristretti. La capacità del bambino di scrivere in modo leggibile, ma anche rapido ed efficiente, gli permette di ottenere sia una comunicazione scritta funzionale che un avanzamento accademico.

La disgrafia, definita come deficit ortografico e nella scrittura a mano, è stata valutata secondo processi neuropsicologici esplicativi all'interno di tre domini primari: elaborazione linguistica, programmazione motoria e cinematica.

Difficoltà di scrittura o disgrafia fu definita da Hamstra-Bletz e Blote come un disturbo nella produzione del linguaggio scritto collegato ai meccanismi di scrittura. È categorizzato come una specifica disabilità di apprendimento. Il problema si manifesta nell'inadeguata prestazione della scrittura tra bambini che sono mediamente intelligenti e che non sono stati identificati come affetti da evidenti problemi neurologici. Inoltre, se la scrittura è molto lenta, i bambini possono dimenticare in quanto la memoria non riesce a mantenere l'informazione per tutto il tempo necessario a trasferirla su carta[2].

Il termine disgrafia è stato abitualmente usato per specificare un disordine dell'espressione scritta del linguaggio durante l'infanzia a cui si oppone, invece, il disordine nella scrittura acquisito in fase adulta che prende il nome di agrafia. In sintesi, la disgrafia indica una scrittura incorretta mentre l'agrafia una mancanza di scrittura[3].

Questa malattia può anche causare altri problemi di apprendimento come: dislessia, deficit di attenzione e disprassia. La disfasia dello sviluppo, la dislessia e la disgrafia sono più comunemente associate nelle famiglie con precedenti storici. Membri diversi della stessa famiglia possono mostrare manifestazioni piuttosto diverse di questi disturbi; per esempio, un bambino disfasico può avere una storia familiare di dislessia e disgrafia specifiche dello sviluppo, ma nessuna disfasia[4].

Il trattamento può variare e include esercizi motori per coordinare la mano con l'occhio. Gli esperti suggeriscono di usare metodi digitali piuttosto che la carta per apprendere. I bambini affetti da disgrafia hanno spesso problemi sociali in quanto si reputano meno capaci dei loro coetanei[5].

L'uso del computer per facilitare la pratica intensiva e ripetitiva si è dimostrato efficace, sia in termini di miglioramento delle valutazioni, sia in termini di benefici funzionali. Viene suggerito l'uso di un dizionario a supporto della strategia e di un elaboratore di testi

adattativo. Il ruolo del computer nella terapia è pensato come strumento per facilitare la pratica ripetitiva e incoraggiarne l'uso indipendente all'interno della terapia[6].

I suoni possono essere utilizzati per informare sulla correttezza di un movimento in corso, senza interferire direttamente con il feedback visivo e propriocettivo[7].

Le prove da recenti studi suggeriscono che la scrittura e il parlato possono essere un aspetto della cognizione capace di identificare specifiche disabilità a pazienti affetti da Parkinson e Alzheimer[8], o la disgrafia spaziale per pazienti con lesioni all'emisfero destro[9].

I.II Diffusione sul territorio

Per stimare il numero il numero di studenti affetti da disturbi specifici dell'apprendimento (DSA), bisogna elaborare i dati sulle certificazioni fornite dal servizio statistico MIUR.

I primi contributi italiani sullo sfondo della prevalenza dei disturbi di letto-scrittura risalgono alla fine degli anni Sessanta. Bisogna sottolineare che i dati appartengono a popolazioni scolastiche difficilmente confrontabili sia per fasce d'età che per territori ed è quindi molto difficile fornire un quadro generale della situazione. In tabella 1 è possibile osservare i dati relativi allo scorso secolo fino al 2001 in cui non abbiamo informazioni sul territorio. In tabella 2 invece abbiamo alcuni dati più recenti, fino al 2015, con informazione sulla regione. Da questi studi sembra emergere che la percentuale di bambini affetti da DSA nelle scuole primarie oscilli tra il 3,2 e il 4,8%[10].

Infine, nel 2016 sono stati raccolti dati su scala nazionale per avere un quadro sulla distribuzione come evidente in tabella 3[11].

Autore e anno	Indicazione nosografica	Campione	Prevalenza %
Faglioni 1967	Dislessia	Primaria	1.34
Cima Venuti 1969	Disgrafia-Dislessia	5° primaria	4.58-5.91
De Renzi 1973	Dislessia	5° primaria	3.6
Levi 1976	Ritardo di lettura	6 < 8 anni	8-10
Bisiacchi 1977	Dislessia	Primaria	0.23-3.05
Guaraldi 1980	Dislessia	6 < 10 anni	1-5
Levi 1981	DSA		3.4
Stella 1987	DSA	Primaria, sec. 1° grado	12-16
Soresi 1993	DSA	3° < 5° primaria	3.6
Levi 1994	Disturbi specifici sviluppo (DSM-III-R)		2.87
Bonanate 1996	DSA	1° primaria	0.7
Tressoldi 1996	DSA		5-10
Stella 1997	Dislessia	Primaria, sec. 1° grado	2.5-3.5
Masala 1998	Probabili DSA	4° primaria	11.8
Coscarella 2001	DSA	Primaria, sec. 1° grado Isola d'Elba	1.5

Tabella 1: Percentuale di bambini disgrafici dal 1967 al 2001

Autore anno	Regione	Popolazione anno scolastico	DSA%	IC95%	N alunni
Barbiero2012	FVG	Primaria 4° as2007/08	3.2	[2.4; 4.3]	1365
Coscarella2015	Elba(LI)	Primaria 3°-5° as2012/13	3.4		827
Cappa 2015	Lanausei(OG)	Primaria 2° as2011/12 as2012/13	4.8	[3.3; 6.7]	610
Costarella2015	Elba(LI)	Secondaria 1° Grado. as2012/13	3.5		792
Coscarella2015	Elba(LI)	Secondaria 2° Grado as2012/13	2.6		1173

Tabella 2: Percentuale di bambini disgrafici dal 2012 al 2015 con la regione

as2015/16	N Alunni	DSA	%DSA
Mezzogiorno	2695740	35465	1.3%
Centro	1381730	50204	3.6%
Nord Est	1347476	46699	3.5%
Nord Ovest	1802170	86779	4.8%
Italia	7227116	219147	3.0%

Tabella 3: Percentuale di bambini disgrafici nel 2016 per zona

In questo caso di studio, presentiamo uno strumento che consente di formulare una diagnosi parziale. Fino ad ora, sembra che non esista in Italia uno strumento che aiuti gli specialisti nella diagnosi di questo disturbo; l'obiettivo è anche quello di far evidenziare una mancanza di supporto automatico per un problema che invece risulta ben presente sul territorio.

I.III Metodi di valutazione

Secondo Ajuriaguerra e colleghi, la disgrafia non è riconducibile ad un disturbo neurologico o ad un limite intellettuale. Essa non viene considerata come un ritardo dello sviluppo ma come uno sviluppo atipico. Sull'esperienza di Ajuriaguerra e dei suoi collaboratori, è stato redatto un elenco di venticinque caratteristiche riguardanti la scrittura dei disgrafici. La tabella 4 mostra l'elenco di tutte le caratteristiche.

La valutazione di una grafia veniva espressa con un punteggio totale, che corrisponde alla somma ponderata dei punteggi ottenuti nelle venticinque caratteristiche riportate dalla scala.

Cattiva utilizzazione dello spazio

D 1	Insieme sudicio e impastoiato
D 2	Rigo spezzato
D 3	Rigo ondulato
D 4	Rigo discendente
D 5	Stretta tra parole
D 6	Spazio irregolare tra parole
D 7	Assenza di margini

Maldestrezza

D 8	Tratto pessimo
D 9	Lettere ritoccate e corrette senza motivo
D 10	Occhielli o asole riempiti di inchiostro
D 11	Torsioni nella zona mediana
D 12	Angolosità nelle arcate
D 13	Suture o saldature
D 14	Collages
D 15	Lettere addossate
D 16	Scrittura a scatti
D 17	Finali slanciati per maldestrezza
D 18	Irregolarità nella dimensione
D 19	Zone mal differenziate
D 20	Lettere atrofizzate

Errori nella forma e nelle proporzioni

D 21	Lettere troppo strutturate o labili
D 22	Forme pessime
D 23	Dimensione troppo piccola o eccessiva
D 24	Cattiva proporzione tra le zone
D 25	Scrittura troppo dilatata o stretta

Tabella 4: Caratteristiche scala Ajuriaguerra

Da alcune ricerche di Hamstra-Bletz è emerso sì che la scala di Ajuriaguerra fosse efficace ma anche che la scala era troppo complessa e la si poteva ridurre selezionando solo alcune voci, in particolare le caratteristiche 3, 8, 9, 16, 18, 19 e 22. A partire da queste, sono state aggiunte le successive mancanti fino ad arrivare alla scala attualmente utilizzata, la BHK (Beknopte Beoordelingsmethode voor Kinderhandschriften). I risultati raccolti con l'una e con l'altra scala hanno mostrato una stretta analogia nonostante il numero di caratteristiche fosse quasi dimezzato.

BHK è stato presentato per la prima volta nei Paesi Bassi nel 1987 dal Dipartimento di Psicologia dello Sviluppo dell'università di Leiden[12]. Attualmente è lo strumento di valutazione della qualità della scrittura più utilizzato nella ricerca e nella pratica clinica. Si tratta di una scala composta da tredici parametri per identificare le caratteristiche dell'atto grafico, che permette l'identificazione della "disgrafia". Il test standard consiste in un semplice esercizio di copia, al fine di facilitare il compito dai componenti di decodifica del

codice alfabetico. Questa metodologia di valutazione permette di identificare e quantificare le caratteristiche salienti della cattiva scrittura.

La sensibilità di un test è data dalla sua capacità di discriminare le minime variazioni relative al parametro misurato, minimizzando in questo modo i falsi negativi. Differenti studi indicano che la BHK è sensibile all'evoluzione dei punteggi relativi all'età e alla presenza o meno di disgrafia. È stata dimostrata, infine, una correlazione dello 0,78 con la scala di Ajuriaguerra. Si è però notato che se la valutazione viene effettuata da operatori non esperti nell'utilizzo della scala, tale indice di correlazione scende a 0,68. Si evidenzia dunque che la scala risente moltissimo dell'esperienza dell'esaminatore stesso e delle procedure di consenso che si stabiliscono intorno alle voci più soggette a variabilità interpretativa.

Per ottenere il punteggio di BHK totale si è proceduto alla somma dei singoli parametri dal primo al tredicesimo. La deviazione standard (DS) del soggetto è il risultato dell'operazione: valore media (M), a seconda del sesso (vedi tabella 5), meno il punteggio totale alla BHK ottenuto dal soggetto diviso il valore della DS, a seconda del sesso (vedi tabella 5).

Il punteggio che pone il soggetto ad una deviazione standard (DS) e mezzo al di sotto della media (-1,5 DS) o un punteggio al di sotto del 10° percentile per sesso (vedi tabella 6), indica una scrittura difficile da leggere e quindi "disgrafica". In presenza di prestazioni leggermente inferiori a questi punteggi ci troviamo di fronte ad una cattiva qualità di scrittura anche se non propriamente disgrafica[12].

	Totale	Maschi	Femmine
Media (M)	19,3	20,3	18,1
Deviazione standard (DS)	6,5	6,1	6,7

Tabella 5: Media e Deviazione standard per sesso

Percentili	Maschi	Femmine
0°	40	39
1°	39	35
5°	33	29
10°	30	26

Tabella 6: Percentili per sesso

Un altro tipo di test è utile e abbastanza diffuso in Italia, sebbene molto recente: la prova di velocità di Tressoldi e Cornoldi. Tale prova, contenuta in una più ampia batteria valutativa, consta di tre compiti di scrittura: la ripetizione del digramma "le", la ripetizione della parola "uno" e la scrittura dei numeri in parola. Si valuta il numero di grafemi scritti in un minuto. Tale prova è standardizzata per tutte le classi della scuola primaria e della scuola secondaria di primo grado. Per quanto riguarda l'aspetto qualitativo, ci si può affidare a delle check list che valutano organizzazione spaziale, qualità del tratto, forme e proporzioni[13].

Per i nostri scopi, verrà utilizzata la scala BHK in quanto consente una buona diagnosi con un numero abbastanza limitato di caratteristiche (o parametri). Attualmente è la scala più utilizzata quando bisogna eseguire diagnosi su bambini fortemente a rischio disgrafia.

Questa scala si è dimostrata uno strumento valutativo sensibile e specifico nell'individuare la qualità di scrittura in corsivo. Ciononostante, il punteggio attribuito a ogni singolo parametro qualitativo del testo è possibile di variazioni relative al giudizio dei singoli valutatori. Questa variabilità può ripercuotersi sul punteggio totale di uno stesso pezzo di scrittura attribuito da due esaminatori differenti, comportando a volte differenze anche significative nel giudizio complessivo finale. Fortunatamente esiste un'ampia fascia di punteggio limite per le scritture povere ma ancora sotto soglia, e solo i punteggi più alti sono classificabili come "disgrafici". In tal modo si sono ridotti notevolmente i casi dubbi[12].

Viene qui presentata una panoramica dei parametri utilizzati dalla scala BHK.

II. Parametri di scrittura

Per la somministrazione del test sono necessari: testo standard da copiare, carta bianca senza righe in formato A4, supporto di scrittura, penna di colore scuro e cronometro. La posizione del bambino deve essere corretta rispetto alla superficie di scrittura.

Il testo standard è un riadeguamento del testo olandese; le prime cinque righe sono: "leo e lo zio", "sono al porto", "mangiano un gelato", "con loro ci sono", "mia e rina". Il bambino dovrà scrivere il testo standard in corsivo sul foglio bianco entro cinque minuti e almeno le prime cinque righe: se non riesce a terminare le prime cinque righe in cinque minuti, gli verrà concesso un tempo supplementare. Per i primi due parametri sono necessari (preferibilmente) tutto il testo copiato; per i successivi solo le prime cinque righe.

Durante il test ci sono alcune caratteristiche che devono essere esaminate separatamente in quanto, se presenti, possono indicare disturbi di carattere generale come per esempio disturbi della vista. Queste caratteristiche possono essere inserite nel paragrafo "Annotazioni". Bisogna prestare attenzione anche alla velocità di scrittura. Un ritmo troppo lento può essere un indizio di problemi in quanto essa può portare a dei crampi che renderanno difficoltoso il processo di scrittura.

La parte più importante di questo sistema consiste nel calcolare il valore dei seguenti tredici parametri di scrittura che permettono la diagnosi: grandezza della scrittura, margine sinistro non allineato, andamento altalenante della linea di scrittura, spazio insufficiente fra le parole, angoli acuti, collegamenti interrotti fra le lettere, collisione fra le lettere, grandezza irregolare delle lettere, misura incoerente fra lettere con e senza estensione, lettere atipiche, forme ambigue delle lettere, lettere ritoccate, traccia instabile. Ogni parametro può avere un valore compreso tra 0 e 5, a differenza del parametro 9, che può arrivare fino a 4.

Va notato che alcuni di questi parametri sono strettamente geometrici mentre altri richiedono un'interpretazione esperta e quindi non tutti i parametri, in questa fase, possono beneficiare di un'analisi completamente automatizzata.

II.I Grandezza della scrittura

La grandezza della scrittura è determinata dall'altezza del corpo delle lettere. Si possono distinguere sette categorie relative alla grandezza che vanno da "più piccolo o uguale a 3 mm" fino a "più grande o uguale a 9 mm". Per ogni esempio di grafia viene indicata la misura delle lettere espressa in millimetri.

La dimensione delle lettere e la classe di appartenenza del bambino determinano il punteggio da assegnare ai fini della valutazione, come indicato nella tabella 7.

	≤ 3 mm	4 mm	5 mm	6 mm	7 mm	8 mm	≥ 9 mm
Classe seconda	0	0	1	2	3	4	5
Classe terza	0	1	2	3	4	5	5
Classe quarta	0	1	2	3	4	5	5
Classe quinta	0	1	2	3	4	5	5

Tabella 7: Punteggi per la grandezza della scrittura

La valutazione deve prendere in considerazione tutto il testo copiato. Il punteggio, che va da 0 a 5, va poi riportato nel foglio di notazione. In Figura 1 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

6 mm (seconda classe primaria, punteggio 2)

Le^s e lo zio
sono al porto
mangiano un gelato
con loro ci sono
mia e rina

Figura 1

II.II Margine sinistro non allineato

Il margine sinistro può non essere allineato verticalmente e presentarsi inclinato verso destra. Nella valutazione viene quantificato, da 1 a 5, il grado di divergenza del testo scritto dal margine verticale del foglio.

Per determinare il punteggio bisogna utilizzare la matrice trasparente (vedi Figura 2) sulla quale sono segnate delle linee oblique contrassegnate da 1 a 5, che indicano le gradazioni di divergenza.

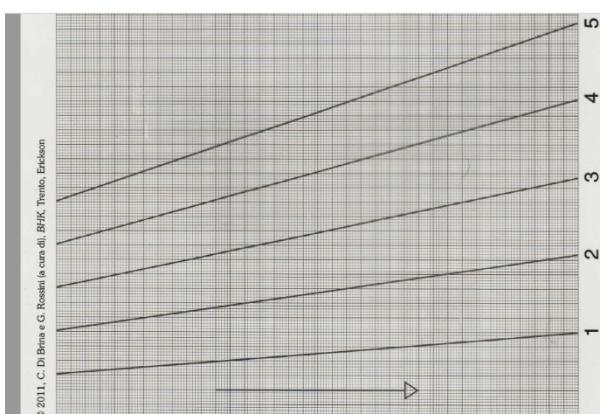


Figura 2: Matrice trasparente per stimare il punteggio esatto

Se la divergenza dal margine è meno marcata della linea 1 della matrice trasparente, si segna 0 sulla scheda di valutazione. Per margini molto irregolari o margini divergenti verso sinistra, va annotato 0 sulla scheda di valutazione. Per margini con divergenza uguale o maggiore della linea 5, bisogna dare il punteggio 5. Quando il margine sinistro segue un andamento alternato, rispettando per più righe un allineamento verticale, si assegna il punteggio 0. Se la tendenza è invece a seguire un andamento divergente, seppure con qualche riga allineata verticale, si quantificherà il grado di divergenza assegnando il punteggio corrispondente. In Figura 3 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

The image shows a handwritten text in Italian on a white background. The text is written in cursive and appears to be a sentence or two. The letters are somewhat slanted and some are positioned above or below a vertical line, illustrating the concept of vertical deviation from a baseline.

le^o e lo zio
sono al po
mangiano
con loro i
mia e rina
Mia è pi
rina inv

Figura 3

II.III Andamento altalenante della linea di scrittura

Questo parametro serve per identificare se la linea di scrittura non è dritta ma ondeggiante. Le lettere all'interno della parola si possono discostare in modo significativo da tale linea, ponendosi al di sopra o al di sotto. Come forme particolari di andamento altalenante si segnala l'andamento interrotto della linea di scrittura e l'andamento alternato.

La valutazione deve prendere in considerazione ciascuna delle prime cinque righe. Per determinare il punteggio, si stabilisce, per ogni riga, se ci sono parole che si discostano significativamente dalla linea di scrittura, ponendosi al di sopra o al di sotto. In caso affermativo, alla riga viene attribuito il valore 1, altrimenti 0. Il punteggio finale andrà da 0

a 5. In Figura 4 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

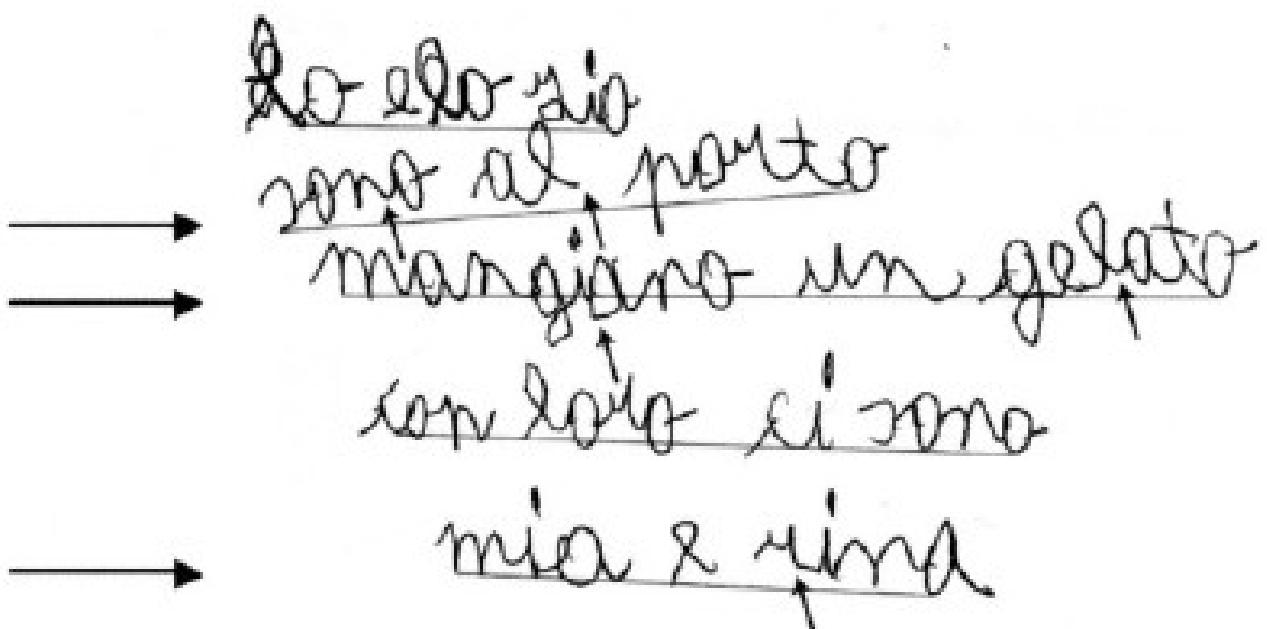


Figura 4

II.IV Spazio insufficiente tra le parole

Lo spazio che separa due parole è considerato insufficiente quando è inferiore alla larghezza del corpo di una lettera (si prende come campione la "o").

La valutazione prende in considerazione ciascuna delle cinque prime cinque righe. La grandezza della "o" si deve rapportare alla grandezza delle lettere che si trovano a sinistra e a destra dello spazio. Se lo spazio fra due parole è inferiore a tale misura, segnare il punteggio 1, altrimenti 0. È sufficiente che la caratteristica sia presente anche una sola volta sulla riga. In caso di dubbio segnare 0; il punteggio finale andrà da 0 a 5.

In Figura 5 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

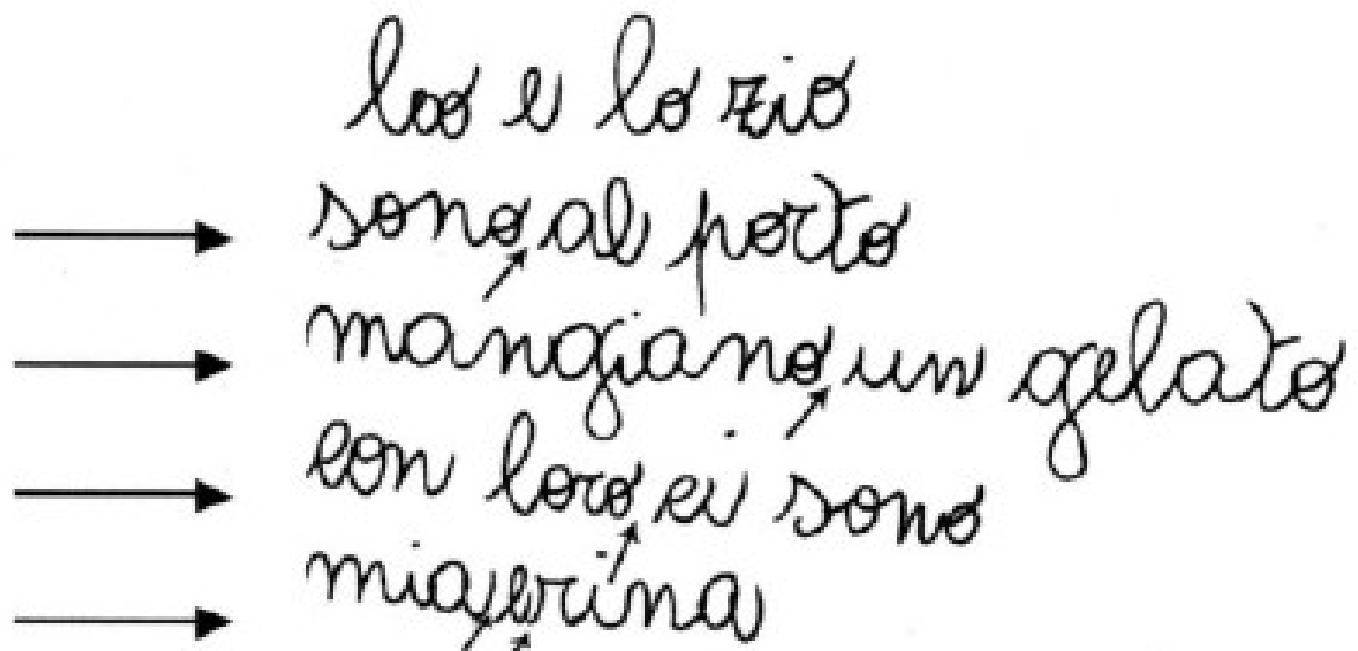


Figura 5

II.V Angoli acuti

Questo parametro viene evidenziato quando la traccia si presenta non fluida, con collegamenti orizzontali allungati e angoli acuti anziché curvi, laddove c'è un cambiamento di direzione. Si tratta di angoli acuti prodotti quando il movimento ascendente cambia in movimento discendente, e viceversa.

La valutazione deve prendere in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano casi di collegamenti orizzontali allungati e/o angoli acuti nei cambiamenti di direzione. In caso affermativo si annota 1, altrimenti 0. È sufficiente che la caratteristica sia presente anche una sola volta sulla riga. In caso di dubbio segnare 0. Il punteggio finale andrà da 0 a 5.

In Figura 6 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

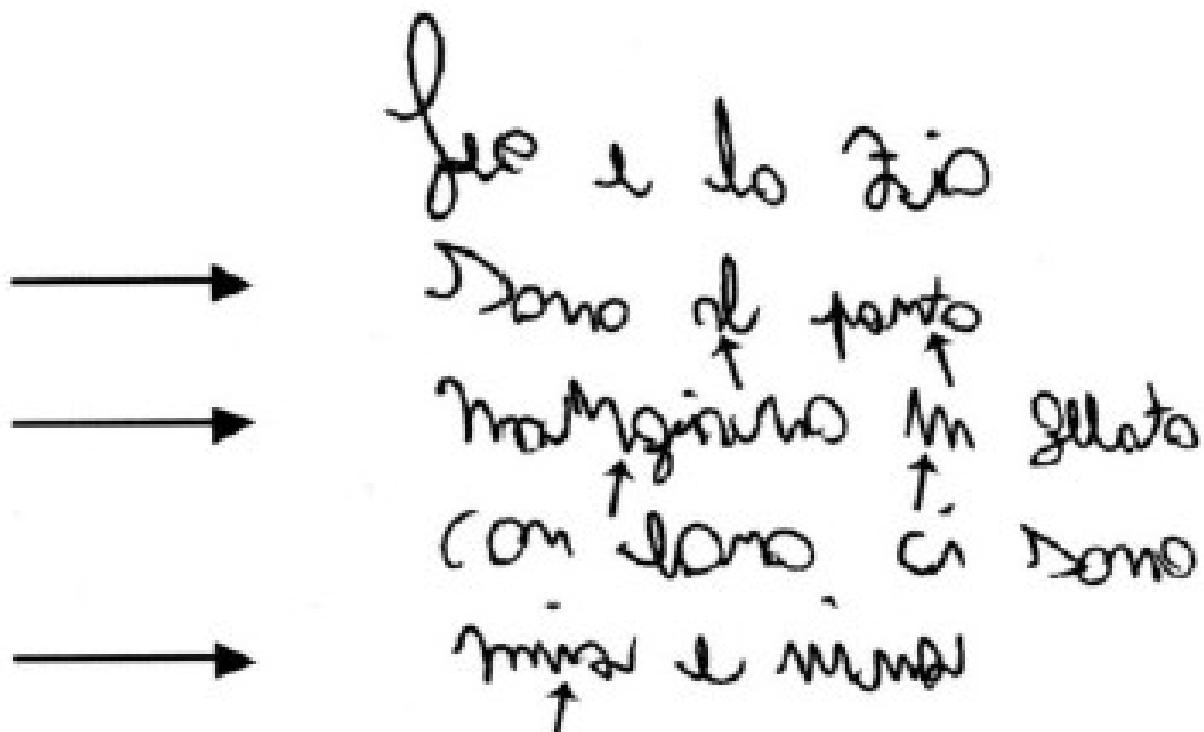


Figura 6

II.VI Collegamenti interrotti fra le lettere

Questo parametro si evidenzia quando il susseguirsi delle lettere all'interno delle parole non avviene con i necessari collegamenti e attraverso un movimento fluido. Il movimento della penna è interrotto nello spazio fra due lettere e a volte la penna viene sollevata dal foglio di scrittura. Tale caratteristica si può manifestare:

- Nell'improvviso cambiamento di direzione del collegamento;
- Nella presenza di punti di maggiore spessore nella traccia di scrittura, o nella presenza di "saldatura";
- Nella mancanza di una linea di collegamento laddove questa avrebbe dovuto esserci;

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano casi di collegamenti interrotti. In caso affermativo si annota 1 sulla scheda altrimenti 0. In caso di dubbio segnare 0. Il punteggio finale andrà da 0 a 5 punti.

In Figura 7 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

leo e lo zio
 → sono al noto
 mangiamo un gelato
 → con loro ci sono
 mia e zia ↑

F

Figura 7

II.VII Collisione fra le lettere

Questo parametro si evidenzia quando lo spazio fra due lettere consecutive è talmente scarso da provocare una collisione fra le lettere o addirittura una parziale sovrapposizione. La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano casi di collisioni fra le lettere secondo la descrizione data qui sopra. È sufficiente che la caratteristica sia presente anche una sola volta sulla riga. In caso di dubbio segnare 0. Il punteggio andrà da 0 a 5.

In Figura 8 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

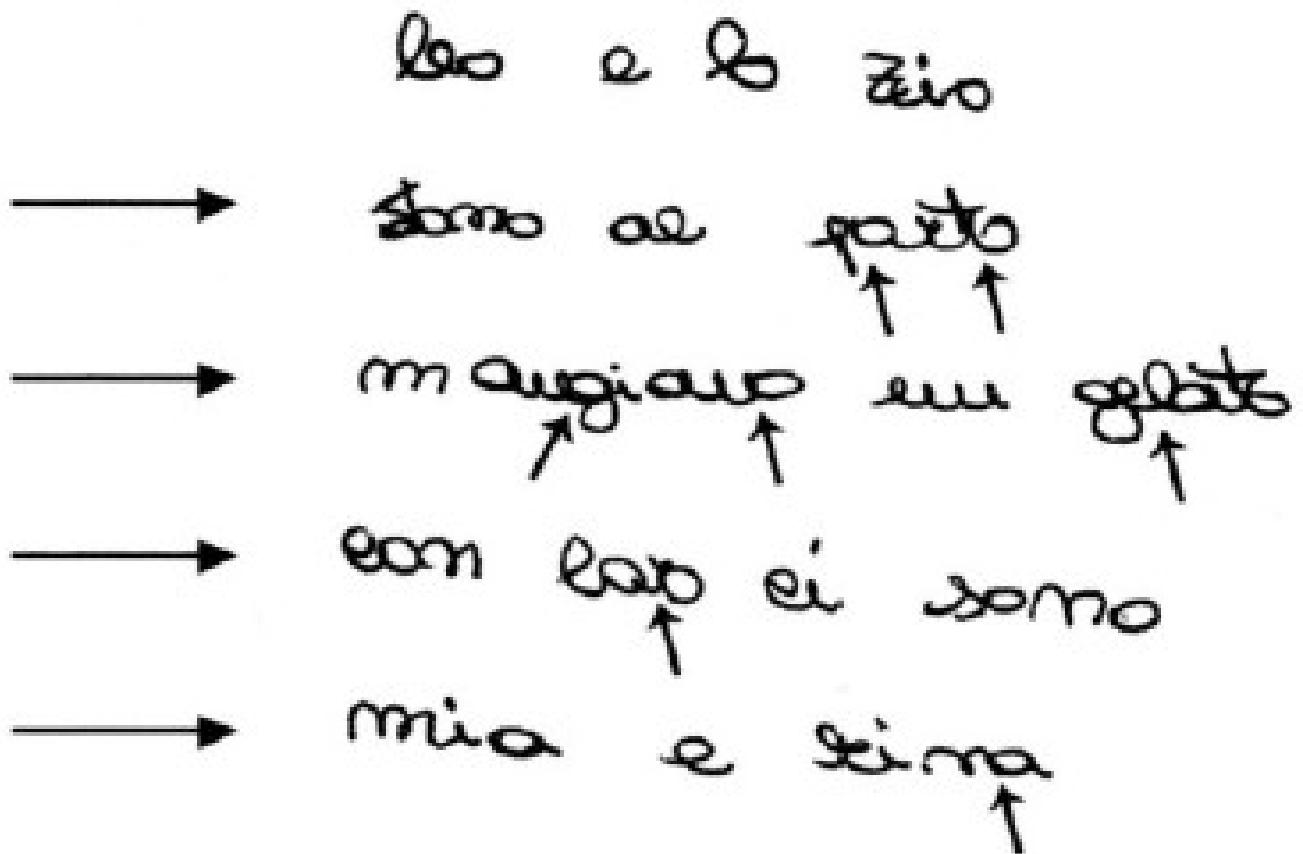


Figura 8

II.VIII Grandezza irregolare delle lettere

Questo parametro si evidenzia quando il corpo delle lettere non ha un'altezza costante. Ciò si manifesta sia nel variare della grandezza del corpo delle lettere all'interno di una parola, sia nel variare della grandezza di intere parole all'interno di una frase.

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase l'altezza in millimetri del corpo della lettera più piccola e di quella più grande. Si individua poi, nelle tabelle 8 e 9, il numero corrispondente alla lettera più piccola, nella colonna di sinistra. Se la misura che abbiamo rilevato del corpo della lettera più grande è uguale o maggiore del numero presente nella colonna di destra, annotare 1 sulla scheda altrimenti 0. In caso dubbio segnare 0. Il punteggio finale andrà da 0 a 5.

In Figura 9 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

Altezza del corpo della lettera più piccola (mm)	Altezza del corpo della lettera più grande (mm)
1,5	2,5
2	3
2,5	4
3	4,5
3,5	5
4	6
4,5	6,5
5	7
5,5	7,5

Tabella 8: Proporzioni accettate tra la lettera più alta e quella più bassa per una singola riga

Altezza del corpo della lettera più piccola (mm)	Altezza del corpo della lettera più grande (mm)
6	8
6,5	8,5
7	9,5
7,5	10
8	11
9	12

Tabella 9: Proporzioni accettate tra la lettera più alta e quella più bassa per una singola riga

les l loro

Tono al porto

Mangiano un gelato

-con loro ci Tono

Mia l rino

Figura 9

II.IX Misura incoerente fra lettere con e senza estensione

Per i nostri scopi è utile definire due gruppi di lettere: le allungate e le non allungate; tra le prime troviamo le seguenti: a, c, e, i, m, n, o, r, s, u, v, w, x, y, z. Le restanti sono non allungate.

Questo parametro si evidenzia se, all'interno delle parole, si distingue a malapena la differenza in altezza di lettere con e senza estensione.

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano delle lettere con estensione la cui altezza si differenzia per niente o pochissimo da quella delle lettere senza estensione. In caso affermativo si annota 1,

altrimenti 0. In caso di dubbio segnare 0. Il punteggio finale andrà da 0 a 4 perché la quinta riga non contiene lettere con estensione.

In Figura 10 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

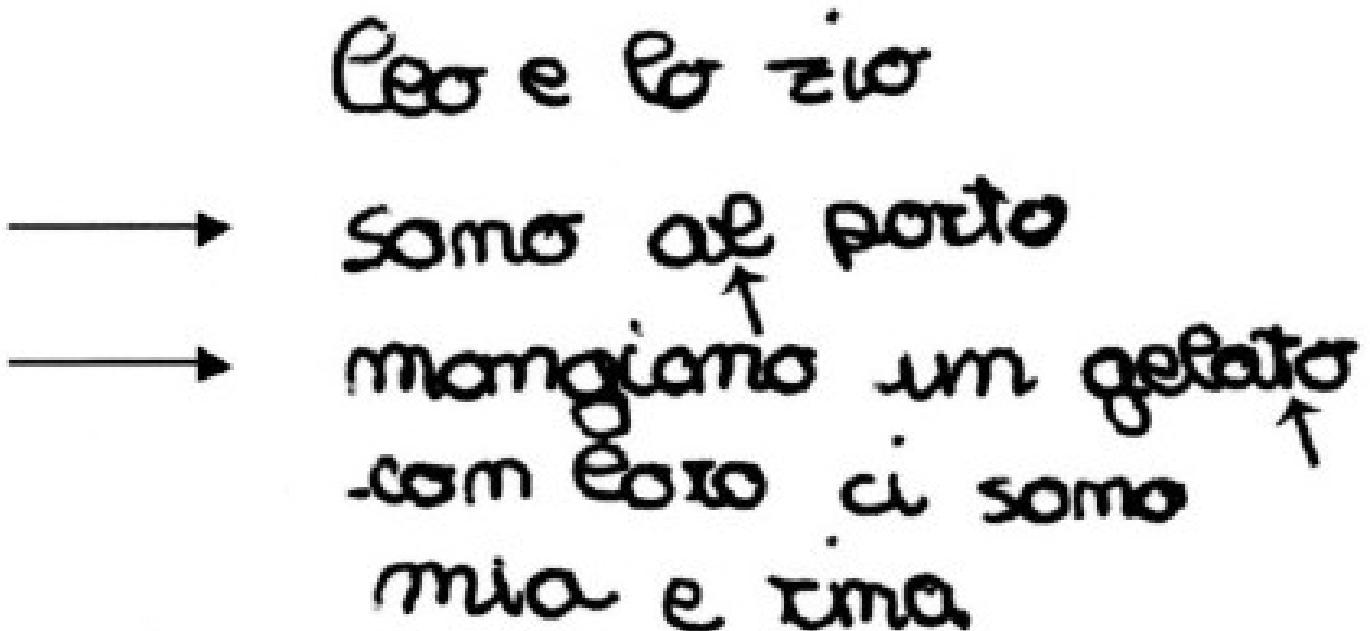


Figura 10

II.X Lettere atipiche

Si definiscono lettere “atipiche” quelle forme di lettere i cui particolari non fanno parte del sistema costruttivo delle lettere alfabetiche. Le forme presentano diversità o deformazioni talmente accentuate da non far più corrispondere la lettera alla forma standardizzata. Per “standardizzata” intendiamo ogni forma che segue un metodo di scrittura riconosciuto.

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano forme atipiche secondo l’analisi dei particolari. In caso affermativo si annota 1, altrimenti 0. In caso di dubbio si segna 0. Il punteggio finale andrà da 0 a 5.

In Figura 11 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

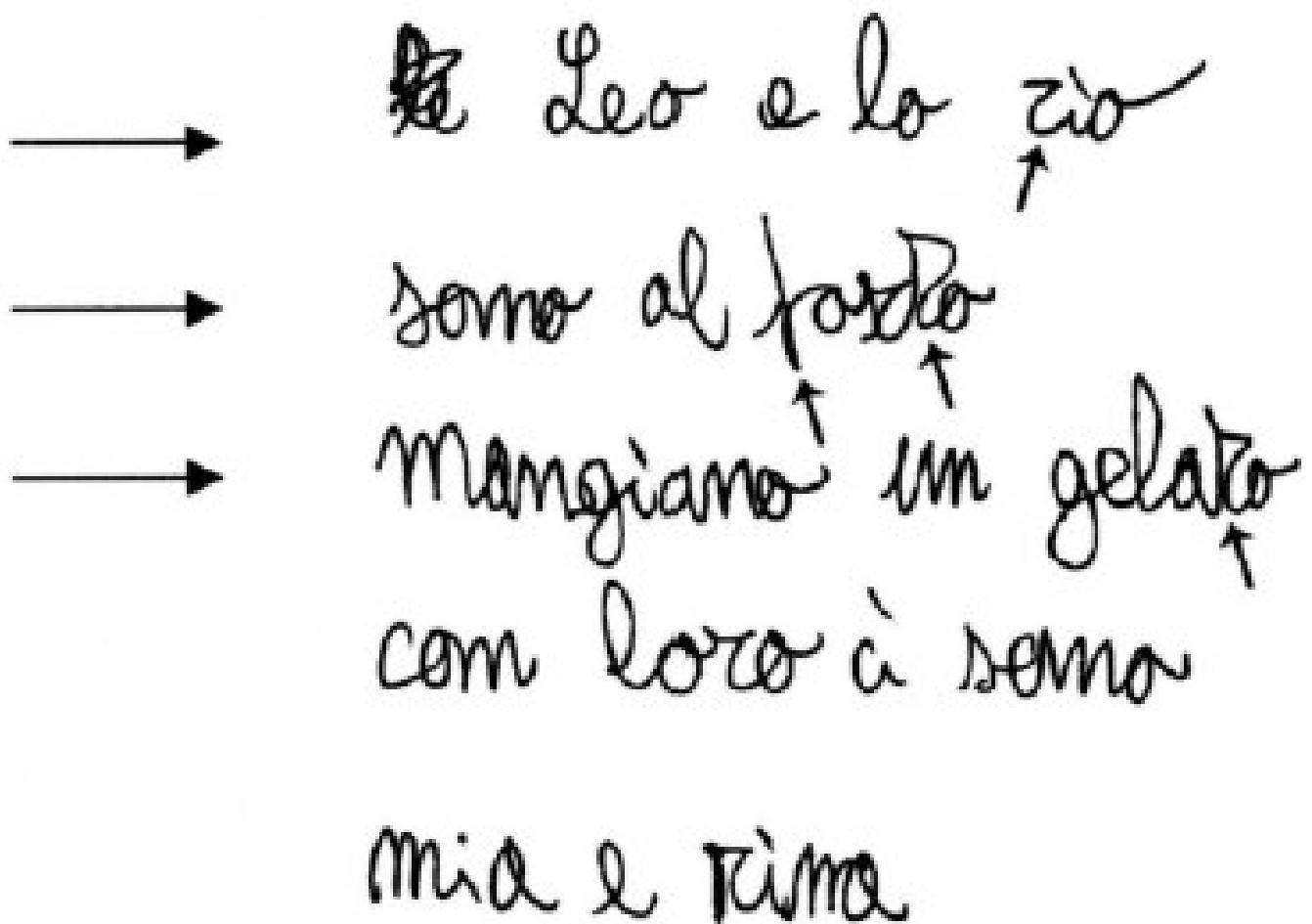


Figura 11

II.XI Forme ambigue delle lettere

Per "forme ambigue delle lettere" si intendono quelle lettere i cui dettagli appartengono alla costruzione delle lettere alfabetiche, ma nelle quali il complesso delle corrispondenze e differenze fra le lettere non viene applicato in modo corretto. Ciò può causare problemi nella corretta interpretazione della lettera. Esempi di possibili cause di ambiguità sono:

- Lettere lasciate aperte che dovrebbero essere chiuse, lettere non complete oppure recanti aggiunte accessorie, una "zampa" di troppo o un tratto troppo prolungato o troppo corto, o inserito all'altezza errata;
- Lettere con asticelle o occhielli troppo lunghi o troppo corti;

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano forme ambigue delle lettere secondo le descrizioni sopra elencate. In caso affermativo si annota 1 altrimenti 0. In caso di dubbio si segna 0. Il punteggio andrà da 0 a 5.

In Figura 12 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

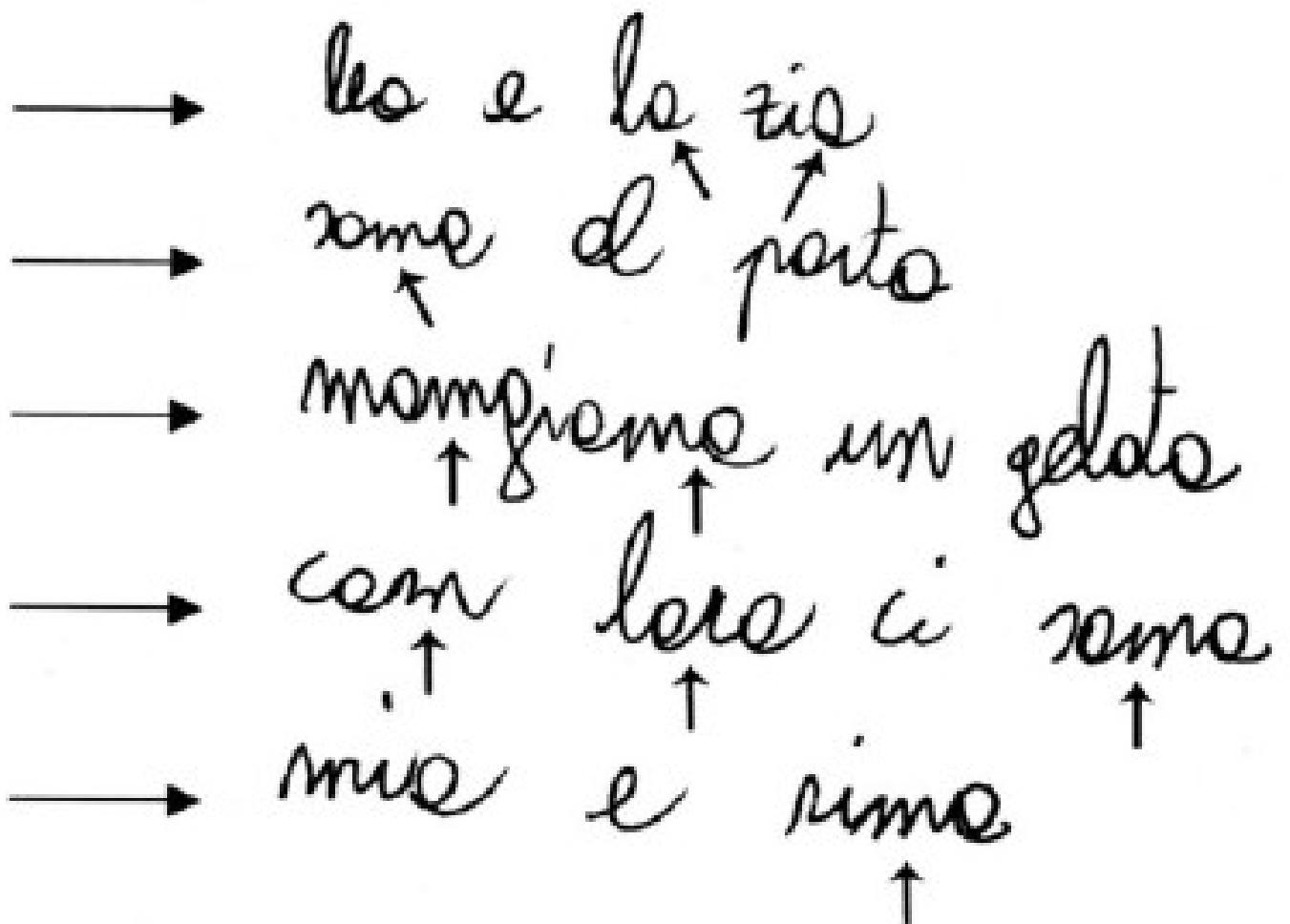


Figura 12

II.XII Lettere ritoccate

Si definiscono lettere ritoccate o ricalcate quelle lettere interamente o parzialmente riscritte, con l'intento di aggiustare e migliorare la loro forma.

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se vi siano lettere ritoccate o ricalcate secondo la descrizione data. In caso affermativo, si annota 1 altrimenti 0. In caso di dubbio si segna 0. Il punteggio andrà da 0 a 5.

In Figura 13 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

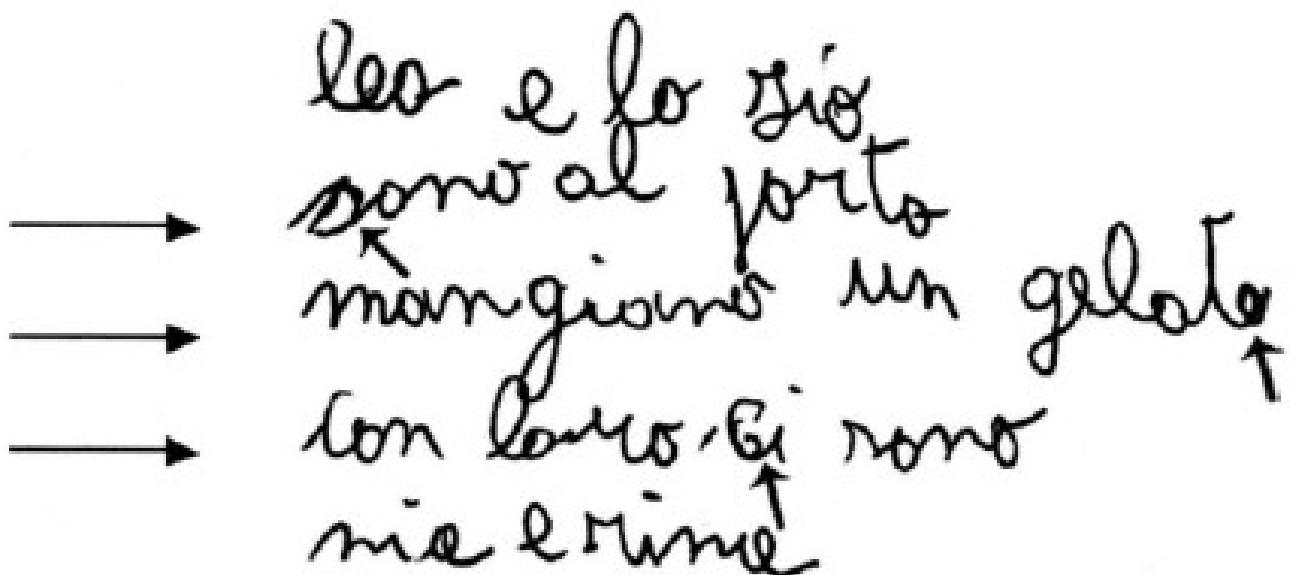


Figura 13

II.XIII Traccia instabile

Si parla di traccia instabile quando la scrittura presenta incertezze, esitazioni e tremolii irregolari. L'irregolarità della traccia è normalmente riscontrabile lungo tutta la traccia di scrittura. La traccia manca di linearità, che è da mettere in relazione con un movimento di scrittura troppo lento e incerto.

La valutazione prende in considerazione ciascuna delle prime cinque righe. Si determina frase per frase se la traccia presenta localmente incertezze e tremolii. In caso affermativo si annota 1 altrimenti 0. In caso di dubbio si segna 0. Il punteggio andrà da 0 a 5.

Nel caso di tremolii regolari parliamo di tremori. Questa caratteristica può essere specificata nella nota "Annotazioni".

In Figura 14 viene mostrato un esempio in cui un bambino presenta problemi con questo parametro.

Leo e lo mio
 sono al porto
 → Mangiamo un gelato
 → Con loro ci sono
 mia e mia

Figura 14

III. Descrizione del sistema

Lo scopo di TestGraphia è quello di stimare automaticamente alcuni parametri e di supportare la facile impostazione di quelli rimanenti, al fine di facilitare la diagnosi; l'esperto deve impostare solo parametri che richiedono interpretazione e/o visione del movimento del bambino. Al termine del processo di valutazione viene generato un report che descrive ciò che il sistema ha rilevato, come le anomalie, i valori dei parametri, il punteggio finale e uno spazio per le annotazioni.

Al testo standard è stata aggiunta una riga con il seguente testo: "a a o o m m" all'inizio (header) e alla fine (footer) del testo. Lo scopo di questa scelta è duplice: per calcolare il primo parametro è necessario eseguire una media dell'altezza delle lettere senza allungamenti e questa riga è fatta ad hoc per questo scopo; la seconda ragione è che per il parametro 4 è necessario conoscere la larghezza media della lettera "o" del bambino e questo è possibile mediando le larghezze delle quattro "o" presenti tra la prima e l'ultima riga. Inoltre, l'intestazione consente al bambino un minimo di riscaldamento per la scrittura del testo effettivo.

III.I Interazione

TestGraphia mostra un pannello iniziale molto semplice, come si può vedere in Figura 15, dove è possibile inserire i dati del bambino e scegliere il file immagine da importare (in formato standard png o jpeg) e il nome del report finale.

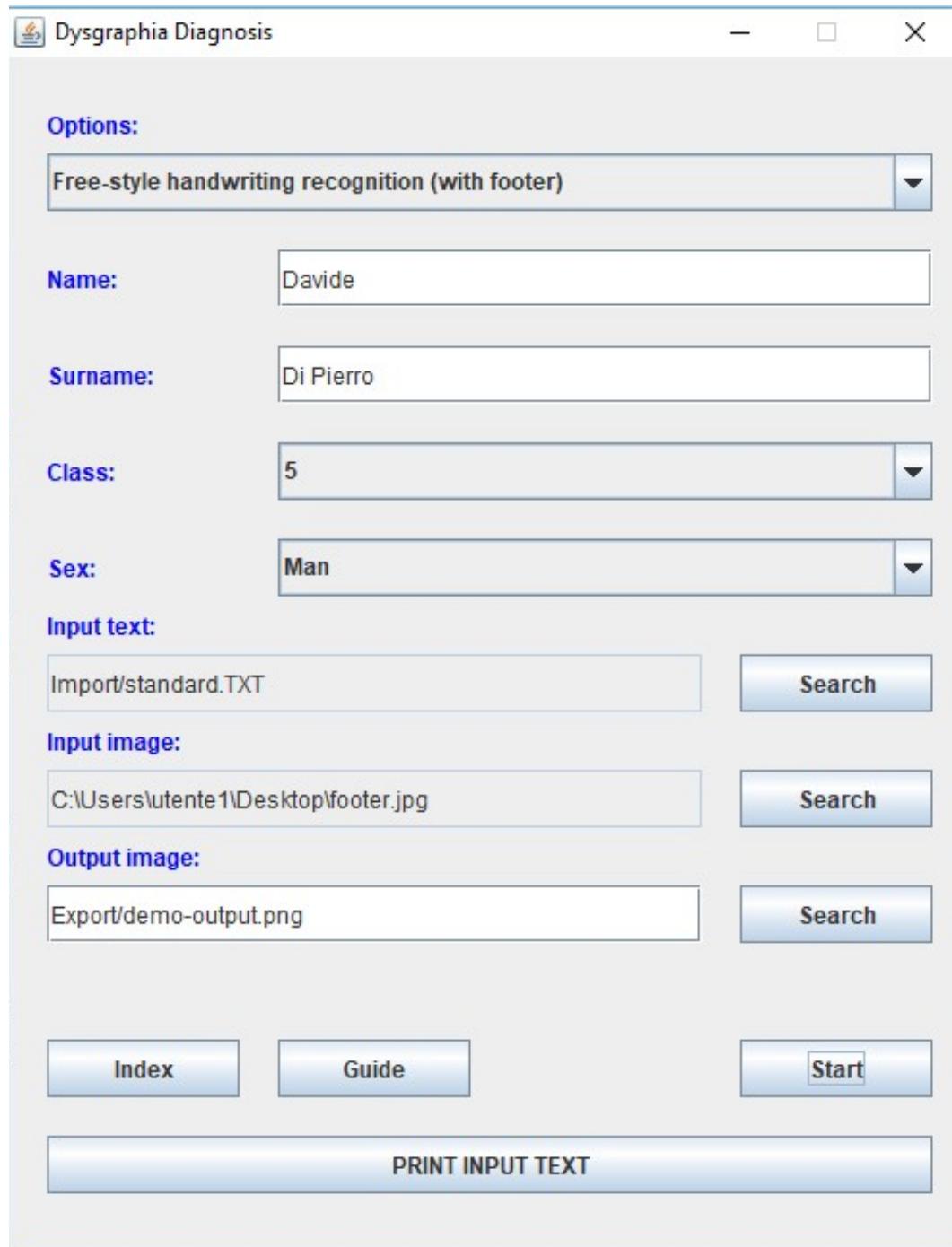


Figura 15: Pannello iniziale in cui si può scegliere: nome, cognome, classe, sesso, immagine di input e nome dell'immagine di output

Cliccando su “Guide” si aprirà un piccolo pdf contenente le istruzioni per utilizzare TestGraphia anche se si è cercato di rendere il più possibile semplice e minimale l’interazione. Cliccando su “Print Input Text” si aprirà un pdf contenente il testo da far ricopiare al bambino sul foglio di carta (vedi Figura 16). Si potrebbe utilizzare anche un altro testo al posto di quello standard ma questa possibilità verrà meglio definita nel paragrafo successivo.

Dysgraphia diagnosis

Text :

a a o o m m

leo e lo zio

sono al porto

mangiano un gelato

con loro ci sono

mia e rina

a a o o m m

Figura 16: Testo standard su pdf

Cliccando su “Start”, dalla Figura 15, si va avanti nell’interazione.

Il parametro 8 è l’unico che viene calcolato in modo semiautomatico: è necessario indicare, per ogni riga, gli estremi della lettera più alta e più bassa. Quindi, dovendo tracciare due punti per lettera e due lettere per linea, chiediamo all’esperto di tracciare venti punti sullo schermo per procedere con l’esecuzione dell’analisi (vedi Figura 17). I punti disegnati vengono evidenziati e viene riportato il numero di punti mancati.

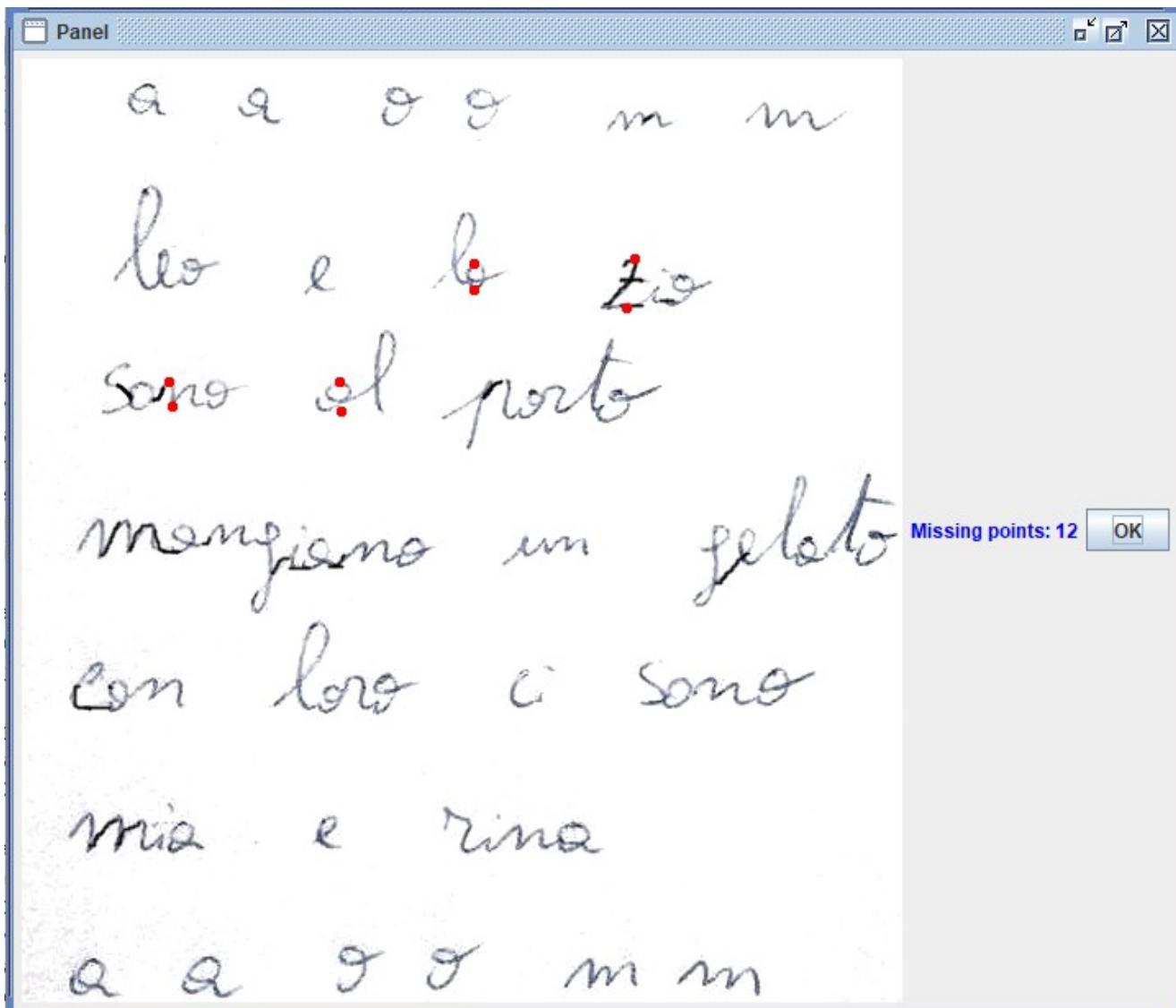


Figura 17: Pannello immagine in cui venti punti devono essere selezionati. Nell'esempio otto sono stati definiti e dodici sono mancanti.

Cliccando su "OK" si prosegue con l'interazione.

Al termine di questa interazione, i parametri 1, 2, 3, 4, 8 e 9 saranno calcolati direttamente dal sistema usando gli algoritmi che andremo a descrivere successivamente. I restanti parametri saranno inseriti dallo specialista usando il pannello mostrato in Figura 18. Per supportare il medico nell'inserimento degli ultimi parametri, viene mostrata l'immagine originale dopo aver applicato algoritmi di text-segmentation come in Figura 19.

Dysgraphia Diagnosis

Name:	Davide		
Surname:	Di Pierro		
Class:	5		
Writing size:	0	Non-aligned left margin:	0
Skewed writing:	1	Insufficient space between letters:	1
Sharp angles:	0	Broken links between letters:	0
Collisions between letters:	0	Irregular size of letters:	2
Inconsistent height between letters:	0	Atypical letters:	0
Ambiguous letters:	0	Traced letters:	0
Unstable track:	0	Finish	

Figura 18: Pannello parametri contenente i parametri coi loro valori. Quelli evidenziati in rosso sono stati calcolati dal sistema (e possono ancora essere modificati) mentre quelli in blu sono a carico dello specialista.

Al termine del processo, i parametri in rosso necessitano sempre di un controllo da parte dell'esperto.

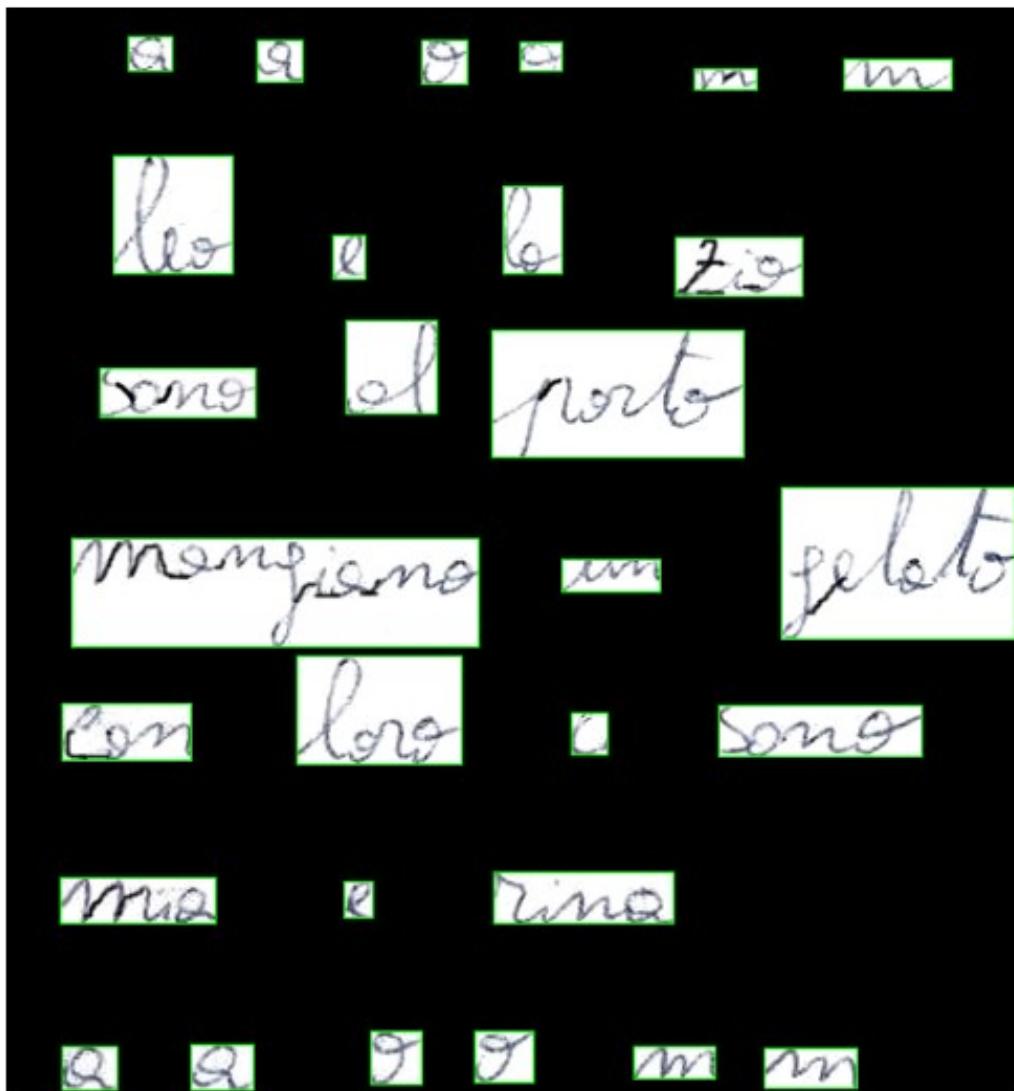


Figura 19: Immagine di input dopo l'elaborazione; lo sfondo bianco è stato sovrascritto dal nero e le parole sono racchiuse in rettangoli verdi

Cliccando su “Finish” (vedi Figura 18), viene generato in formato pdf il report finale del paziente. Su questo documento verranno inserite tutte le informazioni di sintesi della diagnosi. Il report consiste essenzialmente in due fogli: nel primo vengono inseriti, oltre a nome, cognome, classe e sesso del bambino, tutte le anomalie riscontrate dagli algoritmi per ogni parametro calcolato automaticamente. Nel secondo foglio, invece, si realizza una sintesi dei valori dei parametri che consente quindi di ricavare il valore finale da poter confrontare con il valore soglia, che ricordiamo essere -1.5. Viene inoltre stabilito il rapporto tra il punteggio totale e il 10° percentile per sesso che, come ricordiamo, è un altro indicatore di disgrafia. Nelle figure 20 e 21 vengono mostrati i due fogli del report di un paziente.

Dysgraphia diagnosis

Name : Davide
Surname : Di Pierro
Class : 5
Sex : M

1' parameter : Writing size
Average characters' height : 6 mm

2' parameter : Non-aligned left margin
Irregular margin

3' parameter : Skewed writing
Word's baseline : 5.0 mm, Words' average : 5.0 mm, Row's baseline : 5.0 mm
Line 2, inclined baseline

4' parameter : Insufficient space between letters
Line 1, insufficient space between the 1' word and the 2' one
Line 2, insufficient space between the 1' word and the 2' one
Line 3, insufficient space between the 1' word and the 2' one

8' parameter : Irregular size of letters
Line 1, Minimum height : 2.64 mm Maximum height : 5.85 mm
Line 2, Minimum height : 2.45 mm Maximum height : 5.47 mm
Line 3, Minimum height : 2.08 mm Maximum height : 3.77 mm
Line 4, Minimum height : 1.89 mm Maximum height : 5.47 mm
Line 5, Minimum height : 1.51 mm Maximum height : 4.72 mm

9' parameter : Inconsistent height between letters
Percentage : 20 %

Figura 20: Prima pagina del report: dati del bambino con informazioni recuperate dal sistema per ogni parametro calcolato

Dysgraphia diagnosis

Name : Davide
Surname : Di Pierro
Class : 5
Sex : M

1' parameter : Writing size
Value : 3

2' parameter : Non-aligned left margin
Value : 0

3' parameter : Skewed writing
Value : 1

4' parameter : Insufficient space between letters
Value : 3

5' parameter : Sharp angles
Value : 0

6' parameter : Broken links between letters
Value : 0

7' parameter : Collisions between letters
Value : 0

8' parameter : Irregular size of letters
Value : 5

9' parameter : Inconsistent height between letters
Value : 0

10' parameter : Atypical letters
Value : 0

11' parameter : Ambiguous letters
Value : 0

12' parameter : Traced letters
Value : 0

13' parameter : Unstable track
Value : 0

Final score: 1.4 > -1.5
Sum: 12 < 30

Notes:

Figura 21: Seconda pagina del report: dati del bambino con sommario delle metriche, punteggio finale e spazio per le annotazioni e la firma dello specialista.

III.II Personalizzazioni

La diagnosi della disgrafia con l'utilizzo della scala BHK è un metodo abbastanza standardizzato; nulla vieta però di poter modificare delle particolari condizioni per alcuni casi specifici. Le possibili personalizzazioni sono: impostazioni per il parametro 3, impostazioni per il parametro 9, rimuovere il footer e cambiare il testo.

III.II.I Impostazioni per il parametro 3

Per esaminare il parametro 3, andamento altalenante della linea di scrittura, è necessario tener conto di tre valori soglia: media delle parole, baseline della riga e baseline della parola. Il primo serve per identificare un certo scostamento sull'asse y tra le parole della stessa riga, il secondo a rilevare un andamento verso l'alto o verso il basso della riga e il terzo per rilevare lo stesso andamento ma sulla singola parola. La tripla di valori di default

è stata ricavata da prove sperimentali che hanno mostrato che questi valori sono efficaci per il rilevamento di anomalie. Queste soglie, in ogni modo, non sono state prese in comune accordo tra gli esperti e sono quindi soggette a modifiche.

Cliccando su "Index", dalla Figura 15, verranno mostrate le impostazioni predefinite dei parametri, esse possono essere modificate se consentito. Non tutti i parametri rilevanti per la diagnosi possono essere cambiati. Il parametro 3, come riportato in Figura 22, si basa su tre valori modificabili.

Distances	
Words' average	5.0
Row's baseline	5.0
Word's baseline	5.0

Apply **Default**

Figura 22: Pannello indice in cui si possono modificare alcune impostazioni per i parametri, per esempio nel parametro 3 abbiamo le seguenti distanze: media delle parole, baseline riga e baseline parola

III.II.II Impostazioni per il parametro 9

Per esaminare il parametro 9, misura incoerente fra lettere con e senza estensione, è necessario stabilire un rapporto tra l'altezza delle parole che presentano estensioni e quelle che non le presentano. Il valore di default scelto è 20, espresso in percentuale. Questo indica che le parole con allungamenti devono essere alte almeno il 20% in più rispetto a quelle senza. Anche questa soglia è stata ricavata da prove sperimentali e quindi si tratta di un valore non standard. Cliccando su "Index", dalla Figura 15, abbiamo la possibilità di modificare questo parametro inserendo una percentuale che va da 10 a 90, come riportato in Figura 23.



Figura 23: Pannello indice in cui si possono modificare alcune impostazioni per i parametri, per esempio nel parametro 9 abbiamo una soglia percentuale

III.II.III Rimuovere il footer

Come sopra riportato, il testo originale è stato leggermente modificato inserendo la riga “a o o m m” all’inizio e alla fine. La scelta però di inserirla anche alla fine, ha creato dei dubbi in quanto l’ultima riga potrebbe essere scritta con più fatica andando a compromettere lievemente la validità del test. Per questo motivo, si offre la possibilità di avere o meno la riga anche alla fine del testo. Dalla Figura 15, come mostrato in Figura 22, si può scegliere un’opzione di scrittura, di default è previsto il footer.

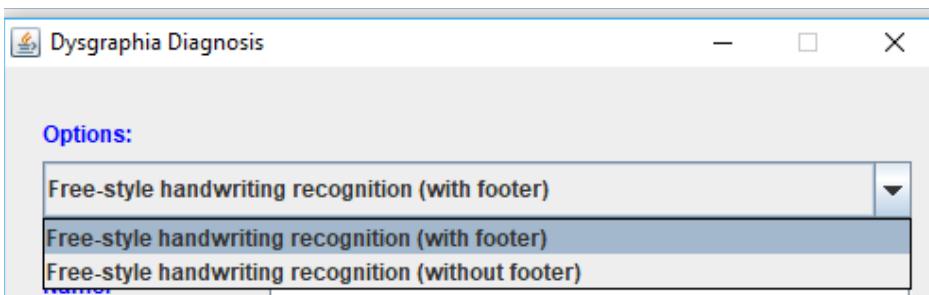


Figura 24: Pannello home con possibilità di modificare l’opzione di scrittura

È bene ricordare che lo scopo di inserire del testo aggiuntivo è legato al calcolo dell’altezza delle parole senza allungamenti. Nel caso in cui ci sia il footer, il parametro 1 terrà conto di header e footer, altrimenti solo dell’header. È anche per questo motivo che si ritiene che la valutazione con footer sia più precisa ed è quindi quella di default.

III.II.IV Cambiare testo

Il testo standard prende spunto da quello olandese originale. Come già detto, è un testo molto semplice per permettere al paziente di non sforzarsi troppo nella comprensione semantica. Esso potrebbe però essere soggetto a delle rivalutazioni nel tempo e possono sorgere motivazioni che inducono al cambiamento. Per anticipare questo bisogno, si dà la possibilità al medico di inserire un testo diverso da quello standard purché rispetti dei vincoli. Per come sono stati concepiti parametri, punteggi e soglie, non si può modificare il numero di righe del testo che quindi deve restare pari a 5. Il parametro 9, che ricordiamo potrà valere da 0 a 4, consiste nell'identificare differenze tra parole con e senza allungamenti. Per questo motivo, tra le cinque righe del testo, esattamente una deve contenere solo parole senza allungamenti. Nel caso del testo standard, l'ultima riga "mia e rina" è quella che non presenta allungamenti. Sempre per questo parametro è poi indispensabile che ogni riga contenga almeno due parole in modo da poter confrontare, per quattro righe su cinque, la differenza in altezza tra parole senza allungamenti e parole con allungamenti.

Esempi di testi non validi sono:

"leo e lo zio
sono al porto
mangiano un gelato
con loro ci sono
mia e laura"

in quanto non ha righe senza allungamenti oppure:

"leo e lo zio
sono a roma
mangiano un gelato
con loro ci sono
mia e rina"

in quanto ha due righe senza allungamenti.

I vincoli possono essere quindi sintetizzati così:

- Testo di cinque righe;
- Allungamenti presenti in quattro righe;
- Almeno due parole per riga;

Stabiliti questi vincoli, si ha poi una grande libertà di scelta. Un testo alternativo a quello standard può essere il seguente:

"leo e mio nonno
sono al parco
mangiano un gelato
con loro non ci sono

mia e nonna”.

Non si dà, invece, la possibilità di modificare il testo aggiuntivo come header ed eventualmente footer, in quanto è stato scelto ad-hoc per uno scopo e non ci sono motivi evidenti per cui debba essere modificato. Per modificare il testo da inserire su TestGraphia, bisogna creare in locale un nuovo file di testo in cui si inserisce il nuovo testo. Dal pannello iniziale in Figura 15 è possibile inserire un file di testo come campo “Input text”. Cliccando su “Start”, verranno eseguiti i controlli sopra descritti e, se il testo è ritenuto valido, partirà la scansione tenendo conto del nuovo file. Cliccando su “Print Input Text” si potrà ora visualizzare su pdf il cambiamento. In Figura 25 viene mostrato un esempio di elaborazione dell’immagine tenendo conto del testo sopra citato.

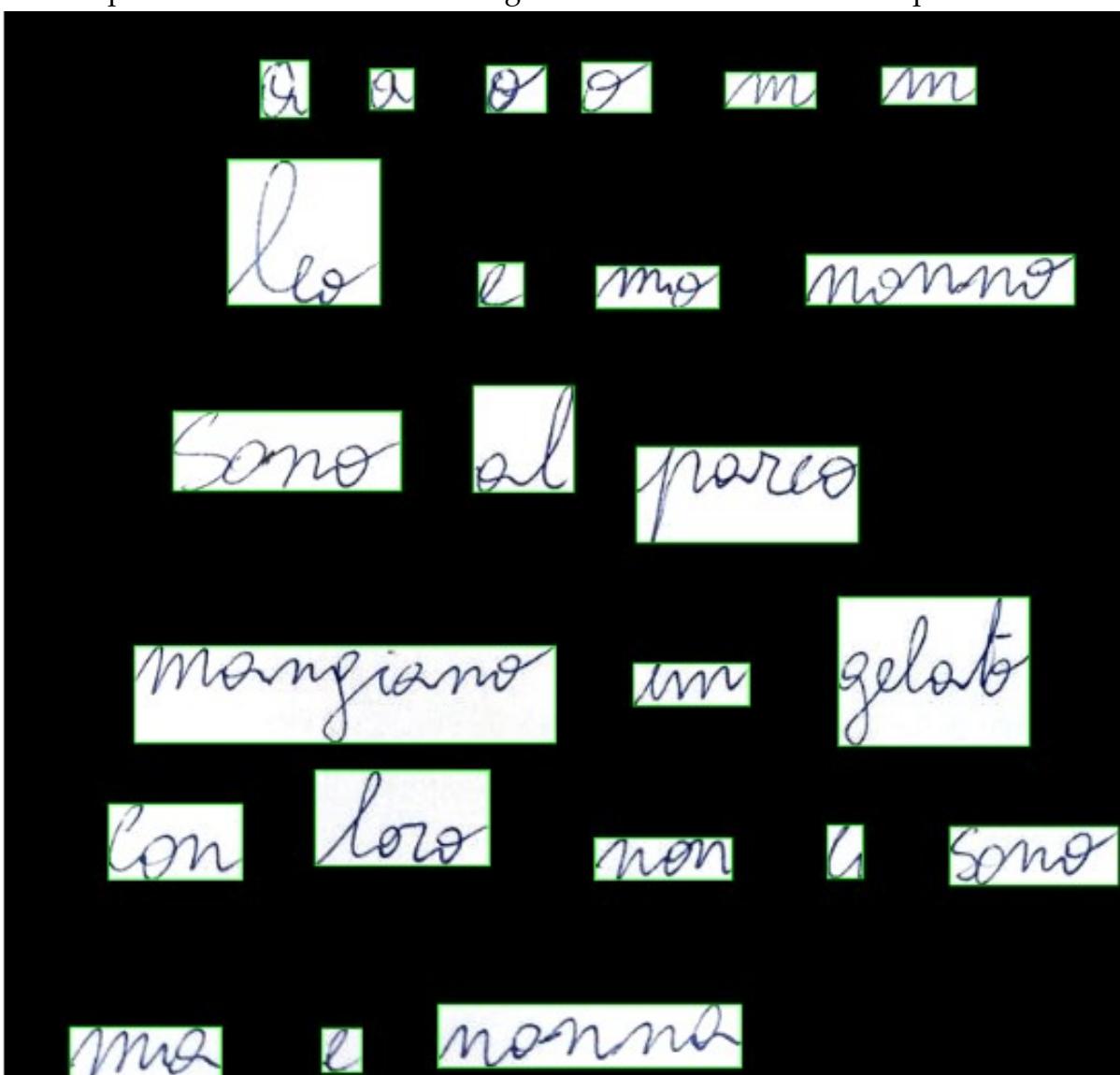


Figura 25: Immagine di input dopo l’elaborazione con testo non standard

È bene ricordare che il testo standard attuale è in uso già da decenni ed è quindi importante valutare con cura ogni possibile modifica.

III.III Algoritmi

Una parte fondamentale per poter iniziare ad elaborare le diverse informazioni del testo scritto è sicuramente la segmentazione delle parole. In letteratura esistono numerose tecniche più o meno efficaci a seconda del tipo di testo che si sta trattando.

La segmentazione di righe di testo scritte a mano non vincolate è difficile a causa della variabilità della distanza tra le righe e della variabilità dell'inclinazione della linea di base. I componenti di due linee di testo consecutive possono essere toccati o sovrapposti in un testo scritto a mano non vincolato. Questi caratteri sovrapposti o toccanti complicano notevolmente il compito di segmentazione delle linee[14]. Esistono metodi che consentono di segmentare il testo in righe senza però dividere le varie parole. Questo approccio si basa sul presupposto topologico che per ogni riga di testo esiste un percorso da un lato dell'immagine all'altro che attraversa una sola riga di testo. Prima si sfoca l'immagine e poi si usano dei tracciatori per seguire i percorsi più bianchi e più neri da sinistra a destra e da destra a sinistra, in modo da suddividere l'immagine in aree di linee di testo[15].

Esiste un approccio basato sulla stima della densità e una tecnica di segmentazione delle immagini all'avanguardia, il metodo del level set. Da un'immagine di un documento di input, stimiamo una mappa di probabilità, dove ogni elemento rappresenta la probabilità che il pixel sottostante appartenga ad una linea di testo. Il metodo del level set viene poi utilizzato per determinare il confine delle linee di testo adiacenti evolvendo una stima iniziale. Estesi esperimenti quantitativi su documenti scritti a mano in stile libero con diversi script, come l'arabo, il cinese, il coreano e l'hindi, dimostrano che il nostro algoritmo supera costantemente i metodi precedenti. Ulteriori esperimenti dimostrano che l'algoritmo proposto è robusto per scalare il cambiamento, la rotazione e il rumore[16].

Nel nostro caso l'algoritmo utilizzato consiste nel costruire istogrammi orizzontali e verticali sulla matrice di pixel. L'istogramma verticale consente di separare le varie parole mentre quello orizzontale le diverse righe.

Il metodo di proiezione orizzontale globale calcola la somma di tutti i pixel neri su ogni riga e costruisce l'istogramma corrispondente. Sulla base dei punti di picco/valle dell'istogramma, le singole linee sono segmentate. Questo metodo presenta alcuni inconvenienti: non funziona su testi obliqui e in situazioni di sovrapposizione[14]. Nonostante queste problematiche, la segmentazione riesce con successo in quanto abbiamo informazioni specifiche sul testo di partenza (sia esso standard o meno) che si possono sfruttare.

III.III.I Segmentazione del testo

Si è partiti da un progetto Java che permette, da un'immagine, di trovare tutte le parole presenti. Le parole vengono riconosciute costruendo la matrice di punti e dividendo le varie parole attraverso istogrammi orizzontali e verticali.

I due problemi principali sono la separazione di parole che hanno sovrapposizioni e l'unione di segmenti diversi che in realtà costituiscono la stessa parola. In entrambi i casi ci aiuta il fatto che sia noto il numero di parole da aspettarci all'interno del documento. Nel caso del testo standard, per esempio, sono ventinove (diciassette di testo più dodici per header e footer). Nel caso in cui due o più parole siano riconosciute come "singola" parola a causa di sovrapposizioni, il numero di parole riscontrate dal sistema non sarà uguale al numero che ci si aspettava. Un esempio di questa situazione è in Figura 26.

a a o o m m
leo e lo zio
Sono al porto
Mangiano un gelato
con loro ci sono
mia e rina

Figura 26: Testo con sovrapposizioni tra "mangiano" e "loro" e tra "gelato" e "sono"

Riconosciuta l'anomalia, si contano quante parole mancano rispetto al numero previsto e si cercano le parole più alte, le candidate a rappresentare due parole unite. Costruendo un istogramma orizzontale sulla matrice di pixel della parola, si trova l'ordinata che rappresenta il punto con meno punti neri. L'ordinata in questione sarà il punto di taglio per separare le parole. Supponiamo di dover dividere in Figura 26 le parole "mangiano" e

“loro”. Dopo aver trovato l’ordinata in cui eseguire il taglio, bisogna tener conto che la lunghezza delle due parole tagliate sarà uguale alla parola più lunga ma è evidente che “loro” non ha la stessa lunghezza di “mangiano”. Per ricalibrare ascissa iniziale e finale delle parole, si cerca il primo punto nero a partire dall’ascissa iniziale originale. Il primo punto nero si trova costruendo un istogramma verticale, la prima ascissa che avrà un numero di punti neri diverso da 0, sarà l’inizio della parola. Partendo dall’ascissa della parola finale, si prosegue in direzione inversa con lo stesso algoritmo per ricavare la fine della parola. Questa tecnica non è ancora sufficiente per una giusta ricalibratura in quanto, come in Figura 26, l’inizio della parola “con” potrebbe essere scambiato come inizio della parola “loro”. Per risolvere questo problema, si verifica una sovrapposizione tra le parole dopo il taglio. Nel momento in cui una parola viene tagliata in due, nel calcolo dell’ascissa iniziale e finale, si rimuove l’area di altre parole presenti in quello spazio in modo che nell’istogramma verticale non risultino punti nell’area della parola “con”. Si verifica quindi che l’interlacciamento non causi errori. Due parole sono interacciate se un estremo sull’asse y di una parola si trova tra i due estremi sull’asse y dell’altra. In termini matematici si verifica che:

$$\begin{aligned} \forall i \in [1, \dots, n] \vee j \in [1, \dots, n] \wedge i \neq j \\ y_{1,j} \leq y_{2,i} \leq y_{2,j} \vee y_{1,i} \leq y_{2,j} \leq y_{2,i} \end{aligned}$$

In questo modo, la divisione è nettamente più accurata. In Figura 27 si nota come sia stata elaborata la Figura 26 dal sistema.

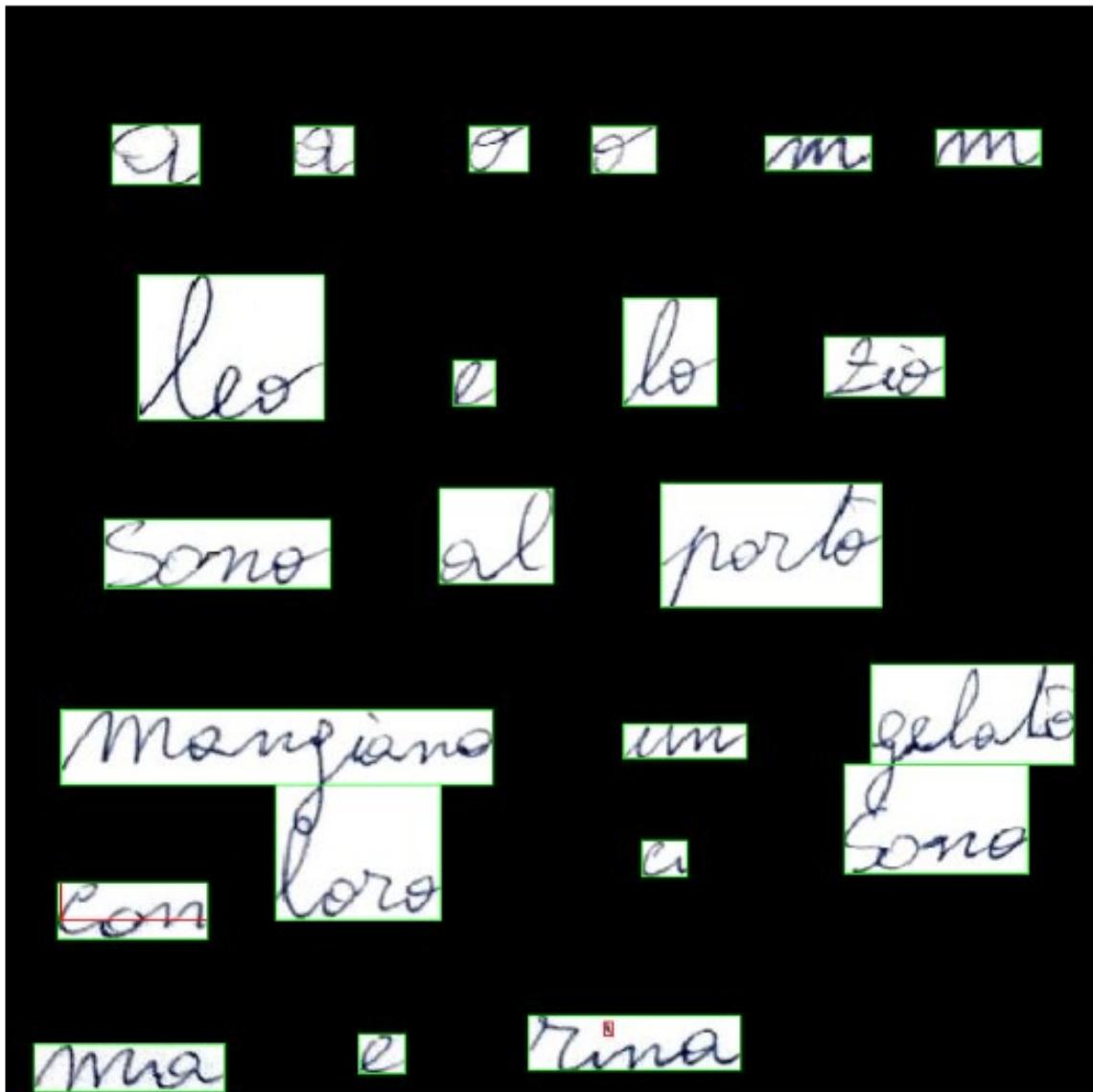


Figura 27: Immagine con parole unite dopo l'elaborazione

Viene qui mostrata l'implementazione per la separazione di un segmento in due parole distinte.

```
private static void separateWords(int[][][] InfoGray) {
    int tallest;
    for (int k = nWords; k < NCHARACTERS + NWORDS; k++) {
        tallest = 0;
        for (int j = 0; j < nWords; j++) {
            if (Words[tallest].y2 - Words[tallest].y1 < Words[j].y2 -
Words[j].y1) {
                tallest = j;
            }
        }
        if (Words[tallest].y2 - Words[tallest].y1 >= 50) {
            int[] count = new int[Words[tallest].y2 - Words[tallest].y1 +
1];
            for (int i = Words[tallest].y1; i <= Words[tallest].y2; i++) {
                count[i - Words[tallest].y1] = 0;
            }
        }
    }
}
```

```

        for (int j = Words[tallest].x1; j <= Words[tallest].x2; j++)
    {
        if (InfoGray[j][i] != 255) {
            count[i - Words[tallest].y1]++;
        }
    }
    int cut = 10;
    for (int j = 10; j < count.length - 10; j++) {
        if (count[cut] > count[j]) {
            cut = j;
        }
    }
    aRect newWord = new aRect();
    newWord.x1 = Words[tallest].x1;
    newWord.x2 = Words[tallest].x2;
    newWord.y2 = Words[tallest].y2;
    newWord.y1 = Words[tallest].y1 + cut;
    Words[tallest].y2 = Words[tallest].y1 + cut;
    adjust(Words[tallest], InfoGray);
    adjust(newWord, InfoGray);
    Words[nWords++] = newWord;
}
}
}

```

Viene qui mostrata l'implementazione per la ricalibrazione di una parola dopo la divisione.

```

private static void adjust(aRect word, int[][] InfoGray) {
    int[] min = new int[word.x2 - word.x1 + 1];
    boolean flagStart = false, flagEnd = false;
    int start = word.x2;
    int end = word.x1;

    for (int i = 0; i < nWords; i++) {
        if (Words[i].x2 > word.x1 && Words[i].x2 < word.x2 &&
areInterlaced(Words[i], word)) {
            if (Words[i].x2 < start) {
                start = Words[i].x2;
                flagStart = true;
            }
        }
    }
    for (int i = 0; i < nWords; i++) {
        if (Words[i].x1 > word.x1 && Words[i].x1 < word.x2 &&
Math.abs(Words[i].y2 - word.y2) < 20) {
            if (Words[i].x1 > end) {
                end = Words[i].x1;
                flagEnd = true;
            }
        }
    }
}

```

```

        }
    }
    if (!flagStart) {
        start = word.x1;
    }
    if (!flagEnd) {
        end = word.x2;
    }
    for (int i = word.x1; i < start; i++) {
        min[i - word.x1] = word.y2;
    }
    for (int i = end + 1; i <= word.x2; i++) {
        min[i - word.x1] = word.y2;
    }
    for (int i = start; i <= end; i++) {
        boolean found = false;
        for (int j = word.y2; j >= word.y1 && !found; j--) {
            if (InfoGray[i][j] < 100) {
                min[i - word.x1] = j;
                found = true;
            }
        }
        if (!found) {
            min[i - word.x1] = word.y2;
        }
    }
    boolean found = false;
    for (int i = min.length - 1; i >= 0 && !found; i--) {
        if (min[i] != word.y2) {
            word.x2 = word.x1 + i;
            found = true;
        }
    }
    found = false;
    for (int i = 0; i < min.length && !found; i++) {
        if (min[i] != word.y2) {
            word.x1 = word.x1 + i;
            found = true;
        }
    }
}
}

```

Il caso opposto si verifica quando vi è un certo spazio tra segmenti che in realtà fanno parte della stessa parola. Anche qui ci viene utile il sapere il numero di parole che ci si aspetta. Se all'interno di una parola, delle lettere presentano spaziatura, il numero di parole riconosciute è maggiore di quello previsto. Attraverso la differenza, sappiamo quante sono le parole da unire. Un esempio di questo caso è proposto in Figura 28.

a a o o m m
ho e ho zio
Sono al porto
Mangiano un gelato
con loro ci sono
Mia e Rina

Figura 28: Testo con segmenti delle parole "zio" e i due "sono" staccati

L'unione delle parole segue una logica puramente basata sulla probabilità ma che non garantisce a pieno il risultato. Si punta sul fatto che se due segmenti sono staccati ma fanno parte della stessa parola, avranno una spaziatura minore rispetto a due parole distinte. Per questo motivo, si uniranno i segmenti più vicini tra loro; verranno bordati di rosso le parole che sono state unite automaticamente dal sistema come in Figura 29.

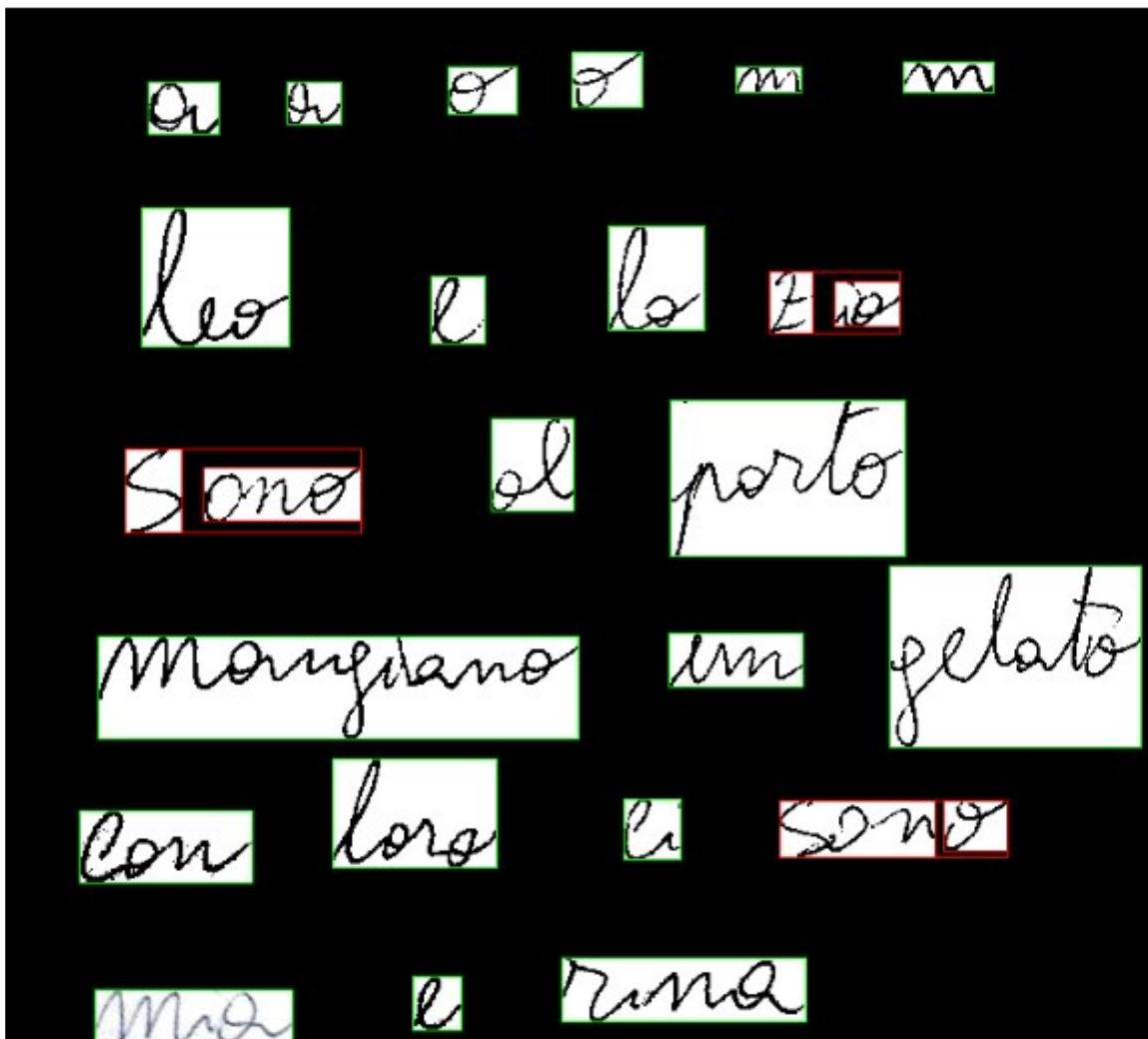


Figura 29: Immagine con segmenti staccati dopo l'elaborazione

Viene qui mostrata l'implementazione per l'unione di segmenti della stessa parola.

```

private static void uniteWords() {
    for (int i = 0; i < nWords - 1; i++) {
        for (int j = i + 1; j < nWords; j++) {
            if(Math.abs(Words[i].x2-Words[j].x1)<15 &&
Math.abs(Words[i].y1-Words[j].y1)<15) {
                Words[i].x2 = Words[j].x2;
                Words[i].y1 = Math.min(Words[i].y1, Words[j].y1);
                Words[i].y2 = Math.max(Words[i].y2, Words[j].y2);
                Words[i].corrected = true;
                for (int k = j; k < nWords - 1; k++) {
                    Words[k] = Words[k + 1];
                }
                nWords--;
            }
        }
    }
}

```

Il problema della segmentazione per righe è di natura più semplice in quanto conosciamo, per ogni riga, il numero di parole. Partendo dall'alto dell'immagine possiamo ordinare le varie parole in base all'ordinata e l'ordinamento ci fornisce già le informazioni sulle righe. Conosciamo, nel caso del testo standard, che le prime sei parole fanno parte dell'intestazione, le successive quattro della prima riga, le successive tre della seconda e così via. In Figura 30 è possibile notare l'andamento delle righe che può essere utile per velocizzare alcuni parametri.

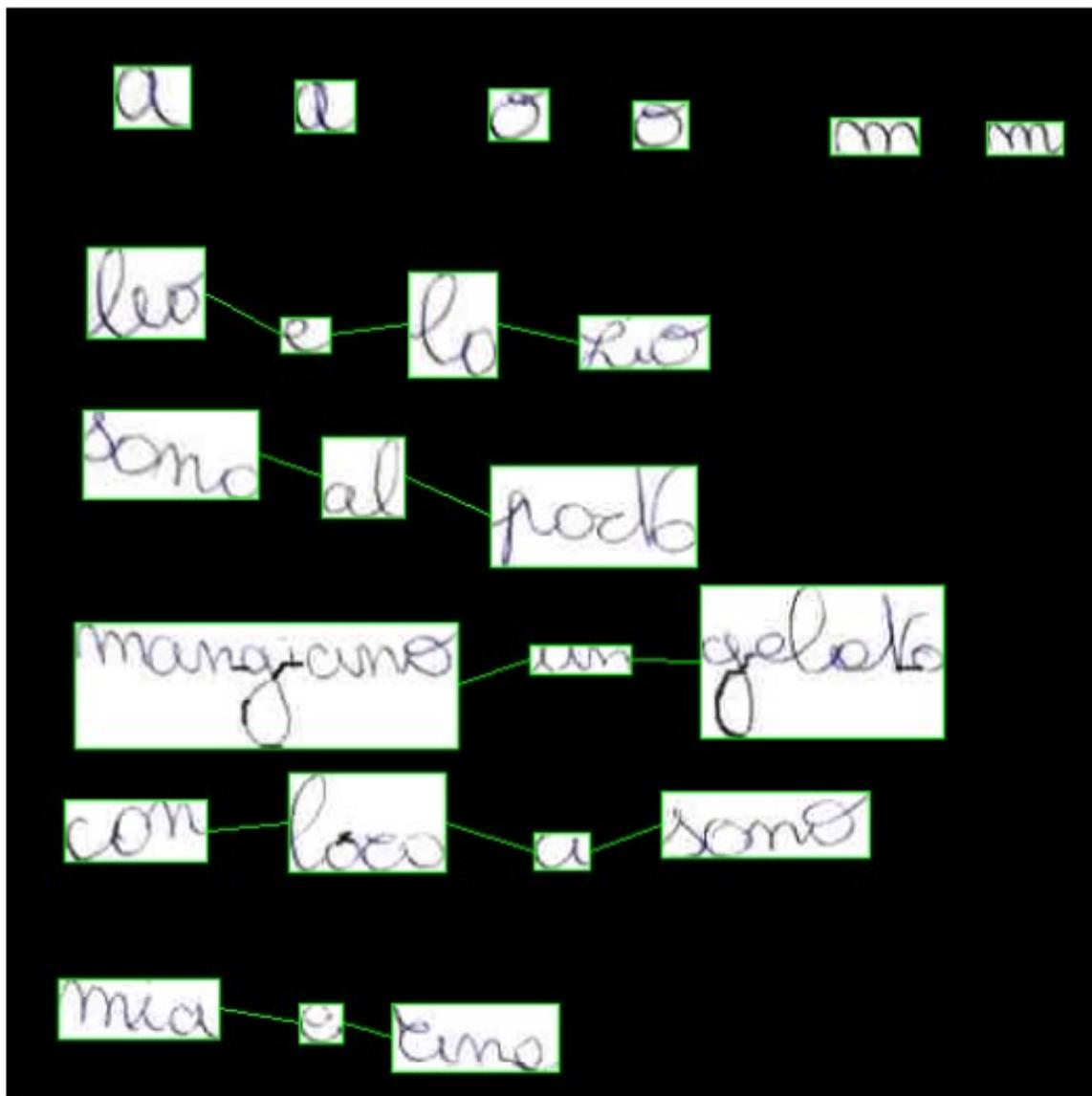


Figura 30: Immagine elaborata con line detection

Viene qui mostrata l'implementazione per la separazione del testo in righe.

```
private static void separateLines() {
    int start;
    for (int i = 0; i < NHEADER; i++) {
```

```

        Header[i] = Words[i];
    }
orderWordsHorizontally(Header, NHEADER);

start = NHEADER;
for (int i = start; i < start + NFIRSTLINE; i++) {
    FirstLine[i - start] = Words[i];
}
orderWordsHorizontally(FirstLine, NFIRSTLINE);
FirstColumn[0] = FirstLine[0];

start += NFIRSTLINE;
for (int i = start; i < start + NSECONDLINE; i++) {
    SecondLine[i - start] = Words[i];
}
orderWordsHorizontally(SecondLine, NSECONDLINE);
FirstColumn[1] = SecondLine[0];

start += NSECONDLINE;
for (int i = start; i < start + NTHIRDLINE; i++) {
    ThirdLine[i - start] = Words[i];
}
orderWordsHorizontally(ThirdLine, NTHIRDLINE);
FirstColumn[2] = ThirdLine[0];

start += NTHIRDLINE;
for (int i = start; i < start + NFORTHLINE; i++) {
    ForthLine[i - start] = Words[i];
}
orderWordsHorizontally(ForthLine, NFORTHLINE);
FirstColumn[3] = ForthLine[0];

start += NFORTHLINE;
for (int i = start; i < start + NFIFTHLINE; i++) {
    FifthLine[i - start] = Words[i];
}
orderWordsHorizontally(FifthLine, NFIFTHLINE);
FirstColumn[4] = FifthLine[0];

start += NFIFTHLINE;
if (footer == 0) {
    for (int i = start; i < start + NHEADER; i++) {
        Footer[i - start] = Words[i];
    }
}
}

private static void orderWordsVertically(aRect[] Words, int lenght) {
    aRect temp;
    for (int i = 0; i < lenght - 1; i++) {
        for (int j = i + 1; j < lenght; j++) {
            if (Words[i].y1 > Words[j].y1) {

```

```

        temp = Words[i];
        Words[i] = Words[j];
        Words[j] = temp;
    }
}
}

private static void orderWordsHorizontally(aRect[] Words, int lenght) {
    aRect temp;
    for (int i = 0; i < lenght - 1; i++) {
        for (int j = i + 1; j < lenght; j++) {
            if (Words[i].x1 > Words[j].x1) {
                temp = Words[i];
                Words[i] = Words[j];
                Words[j] = temp;
            }
        }
    }
}

```

III.III.II Algoritmo grandezza della scrittura

La media delle altezze delle lettere può essere facilmente calcolata utilizzando la prima e l'ultima riga del testo, da noi aggiunte. Nel caso in cui non ci sia il footer, verrà utilizzata solo la prima. Conoscendo le coordinate dei punti estremi di ogni lettera, l'altezza sarà data dalla differenza delle ordinate degli estremi. Una volta ottenuta la media, essa viene arrotondata ad un numero intero e la tabella 7 sopra riportata consentirà di ottenere il valore del parametro.

Viene qui mostrata l'implementazione per il calcolo del valore del parametro 1.

```

private static int calculateHeightCharacters(int clas) {
    double avg = 0;
    int column;
    for (int i = 0; i < NCHARACTERS; i++) {
        avg += Words[i].y2 - Words[i].y1;
    }
    if (footer == 0) {
        for (int i = NWORDS + NCHARACTERS / 2; i < NWORDS + NCHARACTERS; i+
+) {
            avg += Words[i].y2 - Words[i].y1;
        }
    }
    avg /= NCHARACTERS;
    avg = PixelToMm(avg);
    int intAvg = (int) Math.round(avg);
    if (intAvg > 9) {
        column = 6;
    }
}

```

```

    } else if (intAvg < 3) {
        column = 0;
    } else {
        column = intAvg - 3;
    }
    logs[0] = "Average characters' height : " + intAvg + " mm";
    return heightScores[clas - 2][column];
}

```

III.III.III Algoritmo margine sinistro non allineato

Il margine sinistro può essere facilmente calcolato considerando la prima colonna di parole, la più vicina al bordo sinistro escludendo header e footer. Se, scorrendo tra le linee, la distanza dell'asse y dal margine sinistro aumenta, il margine sinistro è inclinato. Per calcolare il valore, si effettua la differenza tra la coordinata x dell'inizio della prima parola della seconda riga e quella dell'inizio della prima parola dell'ultima riga (escluso il footer). Dal foglio mostrato sopra, è possibile misurare la distanza in millimetri e controllare in quale intervallo la distanza si trovi per dare il valore corretto.

Viene qui mostrata l'implementazione per il calcolo del valore del parametro 2.

```

private static int calculateMargin(aRect[] FirstColumn) {
    boolean aligned = true;
    for (int i = 1; i < NROWS && aligned; i++) {
        if (FirstColumn[i].x1 < FirstColumn[i - 1].x1) {
            aligned = false;
        }
    }
    if (!aligned) {
        logs[1] = ("Irregular margin");
        return 0;
    }
    int margin = (int) Math.round(PixelToMm(FirstColumn[NROWS - 1].x1
- FirstColumn[0].x1));
    logs[1] = ("Margin : " + margin + " mm");
    if (margin < 15) {
        return 0;
    }
    if (margin < 27) {
        return 1;
    }
    if (margin < 38) {
        return 2;
    }
    if (margin < 50) {
        return 3;
    }
    if (margin < 62) {
        return 4;
    }
}

```

```
    }  
    return 5;  
}
```

III.III.IV Algoritmo andamento altalenante della linea di scrittura

Durante la scrittura di un documento, l'inclinazione spesso è inevitabile. Poiché gli algoritmi per l'analisi e il riconoscimento di parole e caratteri sono generalmente molto sensibili all'inclinazione, il rilevamento e la correzione delle immagini nei documenti sono passaggi critici. Viene proposto un metodo di rilevamento dell'inclinazione basato sul montaggio in linea retta. Viene introdotto il concetto di "eigen-point", ovvero punto caratteristico all'interno di un'area. Dopo aver analizzato le relazioni tra i successivi eigen-points in ogni riga di testo all'interno di una sottoregione adatta, gli eigen-points posati sulle linee di base vengono selezionati come campioni per il montaggio della linea retta. Viene calcolata la media di queste direzioni di base, che corrisponde al grado di distorsione dell'intera immagine. Gli esperimenti dimostrano che questo approccio è rapido e abbastanza preciso[17].

Una procedura efficiente dal punto di vista computazionale per il rilevamento dell'inclinazione e la determinazione delle righe nei documenti digitalizzati, si basa sulla correlazione incrociata tra i pixel delle linee verticali di un documento. A causa dell'inclinazione del testo, ogni riga di testo orizzontale interseca un insieme predefinito di linee verticali in posizioni non orizzontali. Utilizzando solo i pixel su queste linee verticali costruiamo una matrice e valutiamo l'angolo di inclinazione del documento con grande precisione[18].

Per le immagini di indirizzi complessi viene proposto un approccio di stima dell'angolo di skew basato sull'applicazione di un algoritmo che prende il nome di "Fuzzy runlength". Ogni riga viene scansionata orizzontalmente per contare il numero di pixel neri. Conoscendo il testo che ci si aspetta, si può stimare una soglia di pixel che ci si aspetta di trovare. Se il valore effettivo è minore di quanto stimato, si può concludere empiricamente che il testo presenta inclinazioni. I valori soglia possono anche essere calcolati dinamicamente[19].

Viene proposto un nuovo metodo per rilevare lo skew nelle immagini dei documenti che contengono una combinazione di aree di testo, fotografie, figure, grafici e tabelle. Nel metodo vengono introdotte due idee di base. Un'idea è che serva un nuovo parametro utilizzato per il rilevamento dell'inclinazione per discernere l'orientamento delle righe di testo nelle immagini del documento. Questo parametro si basa sulla complessità dell'immagine del documento e si ottiene dal numero di transizioni da pixel bianchi a pixel neri o viceversa. L'altra idea è che l'inclinazione viene rilevata nelle regioni locali in cui sono previste solo le righe di testo. Tali regioni locali vengono estratte automaticamente da un'immagine del documento e l'angolo di inclinazione ottenuto viene definito come l'inclinazione complessiva del documento. L'inclinazione del documento è

stata misurata in esperimenti con un errore di 0,12 gradi in media per tutti i documenti di prova.[20].

Per i nostri scopi, non abbiamo utilizzato nessuno di questi algoritmi in quanto non forniscono informazioni sull'inclinazione della singola parola e su minimo e massimo della riga. Sono inoltre poco flessibili a dei possibili cambiamenti di soglie.

L'algoritmo di rilevamento dell'andamento altalenante della linea di scrittura ha due diversi livelli: parola altalenante e linea altalenante. Se una linea è altalenante o contiene una parola altalenante, sarà assegnato il valore 1. Il valore del parametro corrisponde al numero di linee in cui è presente un elemento inclinato.

Dopo aver separato il testo nelle varie righe, si esamina, per ogni riga, se ci sono parole altalenanti. Per effettuare questo test, viene creato un vettore di minimi a partire dalla matrice di punti della parola. Il minimo rappresenta l'ordinata del primo pixel nero partendo dal basso della parola. Per verificare se la direzione della parola è coerente, viene calcolata la distanza tra il minimo iniziale della parola e la media dei minimi della parola. Una parola con una direzione coerente ha una differenza al di sotto di una certa soglia, che può essere modificata come mostrato sopra in Figura 22.

Espressa in termini matematici, la proporzione da rispettare è la seguente:

$$\forall i \in [1, \dots, n] \\ \left| m_0 - \frac{1}{n} \sum_{i=1}^n m_i \right| < S_0$$

A livello di linea, viene creato un vettore minimo esteso che considera tutte le parole della linea. Da questo vettore esteso si calcolano le medie dei minimi di tutte le parole. Infine, misuriamo la differenza tra le varie medie per vedere se la linea di base è coerente. Una linea con una direzione coerente presenta una differenza tra le medie al di sotto di una certa soglia, che può essere modificata come mostrato sopra in Figura 22.

Espressa in termini matematici, la proporzione da rispettare è la seguente:

$$\forall i \in [1, \dots, n] \vee j \in [1, \dots, n] \wedge i \neq j \\ \left| \frac{1}{l_i} \sum_{i=1}^{l_i} m_i - \frac{1}{l_j} \sum_{j=1}^{l_j} m_j \right| < S_1$$

Infine, a livello di linea, è anche necessario verificare che l'inizio e la fine non siano troppo distanti. La differenza tra il punto più basso della prima parola e il punto più basso dell'ultima parola viene calcolata dal vettore minimo esteso in valore assoluto. Una linea con un andamento coerente ha una differenza tra i due minimi al di sotto di una certa soglia, che può essere modificata come mostrato sopra in Figura 22.

Espressa in termini matematici, la proporzione da rispettare è la seguente:

$$|m_0 - m_n| < S_2$$

Ai nostri fini, i valori S_0 , S_1 e S_2 sono stati ricavati da prove sperimentali.

Viene qui mostrata l'implementazione per il calcolo del valore del parametro 3.

```
private static int calculateLineSkew(aRect[] Line, int length, int[][]  
InfoGray, int line) {  
    final int maxWidth = 5000;  
    int min[] = new int[maxWidth];  
    int minWords[] = new int[NWORDS];  
    int avg[] = new int[length];  
    int start[] = new int[length];  
    offset = 0;  
  
    boolean skew = false;  
    for (int i = 0; i < length && !skew; i++) {  
        skew = calculateWordsSkew(Line[i], i, min, minWords, avg, InfoGray,  
line);  
    }  
  
    start[0] = 0;  
    for (int i = 1; i < length; i++) {  
        start[i] = start[i - 1] + Line[i - 1].x2 - Line[i - 1].x1;  
    }  
    if (skew) {  
        return 1;  
    }  
  
    int first = min[0];  
    boolean change = false;  
    for (int i = 1; i <= 15 && !change; i++) {  
        if (min[i] != min[i - 1]) {  
            first = min[i];  
            change = true;  
        }  
    }  
    int last = min[offset - 1];  
    for (int i = offset - 2; i >= offset - 15; i--) {  
        if (min[i] > last) {  
            last = min[i];  
        }  
    }  
    if (maxthreshold == 0) {  
        maxthreshold = Index.MAXTHRESHOLD;  
    }  
    if (Math.abs(first - last) > maxthreshold) {  
        logs[2] += ("Line " + line + ", inclined baseline\n");  
        return 1;  
    }  
    boolean bug = false;  
    int minRelative = Math.max(first, last);  
    for (int i = 1; i < length - 1; i++) {  
        if (minRelative - minWords[i] > ABOVETHRESHOLD) {
```

```

        bug = true;
    }
}
for (int i = 0; i < length - 1 && !bug; i++) {
    for (int j = i + 1; j < length && !bug; j++) {
        if (wordsthreshold == 0) {
            wordsthreshold = Index.WORDSTHRESHOLD;
        }
        if (Math.abs(avg[i] - avg[j]) > wordsthreshold) {
            logs[2] += ("Line " + line + ", variable words' average\n");
            bug = true;
        }
    }
}
if (bug) {
    return 1;
}
return 0;
}

private static boolean calculateWordsSkew(aRect word, int index, int[] min,
int[] minWords, int[] avg, int[][][] InfoGray, int line) {
    avg[index] = 0;
    minWords[index] = 0;
    int length = word.x2 - word.x1 + 1;
    for (int i = word.x1; i <= word.x2; i++) {
        boolean found = false;
        for (int j = word.y2; j >= word.y1 && !found; j--) {
            if (InfoGray[i][j] < 100) {
                min[offset++] = j;
                if (minWords[index] < j) {
                    minWords[index] = j;
                }
                avg[index] += j;
                found = true;
            }
        }
        if (!found) {
            min[offset++] = word.y2;
            avg[index] += word.y2;
        }
    }
    avg[index] /= length;
    if (avgthreshold == 0) {
        avgthreshold = Index.AVGTHRESHOLD;
    }
    if ((Math.abs(min[offset - length] - avg[index])) > avgthreshold) {
        logs[2] += ("Line " + line + ", skewed word\n");
        return true;
    }
    return false;
}

```

}

III.III.V Algoritmo spazio insufficiente fra le parole

La misura della larghezza della lettera "o" è facilmente determinabile calcolando la media della larghezza (differenza tra l'ascissa finale e quella iniziale) delle due "o" presenti nella prima riga e delle due "o" presenti nell'ultima riga, se presente. Per ogni riga, quindi, verifichiamo se ci sono due parole che sono distanti meno della larghezza media della "o". La distanza tra due parole w e (w+1) è data semplicemente dalla differenza tra l'ascissa iniziale della parola (w+1) e l'ascissa finale della parola w.

Il valore del parametro sarà semplicemente il numero di righe in cui due parole sono molto inferiori alla larghezza media della "o".

Viene qui mostrata l'implementazione per il calcolo del valore del parametro 4.

```
private static int calculateDistanceWords() {
    boolean found = false;
    int count = 0;
    logs[3] = "";
    for (int i = 1; i < NFIRSTLINE && !found; i++) {
        if (FirstLine[i].x1 - FirstLine[i - 1].x2 < oWidth) {
            logs[3] += ("Line 1, insufficient space between the " + i
+ "' word and the " + (i + 1) + "' one\n");
            found = true;
            count++;
        }
    }
    found = false;
    for (int i = 1; i < NSECONDLINE && !found; i++) {
        if (SecondLine[i].x1 - SecondLine[i - 1].x2 < oWidth) {
            logs[3] += ("Line 2, insufficient space between the " + i
+ "' word and the " + (i + 1) + "' one\n");
            found = true;
            count++;
        }
    }
    found = false;
    for (int i = 1; i < NTHIRDLINE && !found; i++) {
        if (ThirdLine[i].x1 - ThirdLine[i - 1].x2 < oWidth) {
            logs[3] += ("Line 3, insufficient space between the " + i
+ "' word and the " + (i + 1) + "' one\n");
            found = true;
            count++;
        }
    }
    found = false;
    for (int i = 1; i < NFORTHLINE && !found; i++) {
```

```

        if (ForthLine[i].x1 - ForthLine[i - 1].x2 < oWidth) {
            logs[3] += ("Line 4, insufficient space between the " + i
+ "' word and the " + (i + 1) + "' one\n");
            found = true;
            count++;
        }
    }

    found = false;
    for (int i = 1; i < NFIFTHLINE && !found; i++) {
        if (FifthLine[i].x1 - FifthLine[i - 1].x2 < oWidth) {
            logs[3] += ("Line 5, insufficient space between the " + i
+ "' word and the " + (i + 1) + "' one\n");
            found = true;
            count++;
        }
    }
    return count;
}

```

III.III.VI Algoritmo grandezza irregolare delle lettere

Come abbiamo detto, per questo parametro è necessario avere un'interazione con l'utente che indicherà, per ogni riga, gli estremi della lettera senza allungamento massimo e minimo. Il sistema esaminerà le coppie di punti per linea ed eseguirà la differenza tra le ordinate. Riconoscendo, per ogni riga, quale delle due lettere è più alta, utilizzerà le tabelle 8 e 9 sopra riportate per assegnare il valore 1 o 0. Il valore del parametro sarà il numero di righe in cui l'altezza tra la lettera più alta e quella più bassa è incoerente.

Viene qui mostrata l'implementazione per il calcolo del valore del parametro 8.

```

private void calculateHeightScore() {
    for(int i=0; i<NROWS; i++) {
        lines[i] = new LineForHeight();
        calculateLineHeightScore(lines[i], vectorY[i*4],
vectorY[i*4+1], vectorY[i*4+2], vectorY[i*4+3]);
    }
}

private void calculateLineHeightScore(LineForHeight line, int y1, int y2,
int y3, int y4) {
    int h1 = Math.abs(y1-y2);
    int h2 = Math.abs(y3-y4);
    if(h1 > h2) {
        line.maxHeight = h1;
        line.minLength = h2;
    }
}

```

```

}else {
    line.maxHeight = h2;
    line.minLength = h1;
}
line.maxHeight = Math.round(DotsToMm(line.maxHeight)*100)/100.0;
line.minLength = Math.round(DotsToMm(line.minLength)*100)/100.0;
line.score = 0;
if(line.maxHeight<=3 && line.maxHeight-line.minLength>1) {
    line.score = 1;
}
if(line.maxHeight<=5 && line.maxHeight-line.minLength>1.5) {
    line.score = 1;
}
if(line.maxHeight<=8.5 && line.maxHeight-line.minLength>2) {
    line.score = 1;
}
if(line.maxHeight<=10 && line.maxHeight-line.minLength>2.5) {
    line.score = 1;
}
if(line.maxHeight-line.minLength > 3) {
    line.score = 1;
}
}
}

```

III.III.VII Algoritmo misura incoerente fra lettere con e senza estensione

Per ogni riga misuriamo l'altezza delle parole con allungamenti e l'altezza delle parole senza allungamenti. Se c'è almeno un rapporto tra parole con e senza allungamenti che non rispetta la proporzione, il valore 1 sarà assegnato. Questa percentuale, come già detto, può essere scelta dallo specialista dal pannello Index mostrato in Figura 23. Il valore del parametro rappresenta il numero di righe in cui la proporzione non viene rispettata; il valore può arrivare fino a 4 perché nell'ultima riga (nel testo standard) non ci sono parole con allungamenti.

Viene qui mostrata l'implementazione per il calcolo del valore del parametro 9.

```

private static int calculateStretch(aRect[] FirstLine, aRect[] SecondLine,
aRect[] ThirdLine, aRect[] ForthLine, aRect[] FifthLine) {
    logs[4] = "";
    if (ratethreshold == 0) {
        ratethreshold = Index.RATETHRESHOLD;
    }
    logs[4] += "Percentage : " + Math.round((ratethreshold - 1) * 100) + "%\n";
    return detectStretch(FirstLine, setFirst, 1) + detectStretch(SecondLine,
setSecond, 2) + detectStretch(ThirdLine, setThird, 3)
}

```

```

        + detectStretch(ForthLine, setForth, 4) +
detectStretch(FifthLine, setFifth, 5);

    }

    private static int detectStretch(aRect[] Line, HashSet<Integer> set, int
line) {
    List<Integer> heightNormalCharacters = new LinkedList<>();
    List<Integer> heightLongCharacters = new LinkedList<>();

    for (int i = 0; i < Line.length; i++) {
        if (set.contains(i)) {
            heightLongCharacters.add(Line[i].y2 - Line[i].y1);
        } else {
            heightNormalCharacters.add(Line[i].y2 - Line[i].y1);
        }
    }

    for (int heightNormal : heightNormalCharacters) {
        for (int heightLong : heightLongCharacters) {
            double rate = (double) heightLong / (double) heightNormal;
            if (ratethreshold == 0) {
                ratethreshold = Index.RATETHRESHOLD;
            }
            if (rate < ratethreshold) {
                logs[4] += "Line " + line + ", inconsistent height of words\
n";
                return 1;
            }
        }
    }
    return 0;
}

```

IV. Risultati

IV.I Progettazione test

Per verificare la validità del nostro sistema software, sono stati raccolti i manoscritti dei bambini dal secondo al quinto anno di scuola primaria. Anche se non disponiamo di informazioni precise sul numero di maschi e sul numero di femmine, è noto che i campioni sono stati prelevati in modo omogeneo. Per alcuni individui, abbiamo anche informazioni sul sesso che ci permettono di fare una diagnosi completa. In totale sono stati raccolti 21 campioni. Per la raccolta, gli insegnanti della scuola primaria hanno offerto il loro aiuto per sottoporre questo test ai loro studenti. Fornite tutte le informazioni necessarie di cui al paragrafo 2, hanno poi raccolto e consegnato i manoscritti con l'indicazione della classe e, se l'insegnante lo permetteva, anche del sesso. Dopo la raccolta,

si è constatato che 5 manoscritti non erano validi ai fini del test e sono stati scartati per i seguenti motivi: mancanza di intestazione, scrittura in lettere maiuscole o non conformità con il testo standard fornito.

I 16 campioni raccolti sono riassunti nella tabella 10.

Sample	Males	Females	Not specified	Total
Affected by DSA	2	0	1	3
Not affected by DSA	3	3	7	13
Total	5	3	8	16

Tabella 10: L'intero campione con informazioni sul sesso e sul numero di bambini affetti da DSA.

IV.II Validazione

Dopo la raccolta, per ogni bambino, sono stati calcolati manualmente i parametri che il sistema ha il compito di identificare (1, 2, 3, 4, 8 e 9). I risultati ottenuti sono stati raccolti nella tabella 11.

Manually	Paramether 1	Paramether 2	Paramether 3	Paramether 4	Paramether 8	Paramether 9
2,1	0	0	1	1	3	1
2,2	1	0	0	5	4	1
2,3	0	0	0	5	3	0
2,4	1	0	3	5	2	0
2,5	2	0	0	0	5	2
2,7	2	0	3	4	5	3
2,8*	4	0	2	3	4	2
3,1*	2	0	2	5	5	1
3,3	2	0	2	4	4	1
3,4	3	0	1	2	5	1
3,5*	3	1	1	5	5	2
3,6	2	0	1	4	5	0
3,7	3	0	2	3	4	0
3,8	2	0	0	5	4	1
4,1	2	0	0	2	5	3
5,3	2	0	1	0	5	0

Tabella 11: I parametri 1, 2, 3, 4, 8 e 9 sono stati identificati manualmente per ogni bambino. Il primo numero della prima colonna indica la classe, mentre l'asterisco indica che il bambino è affetto da DSA.

Successivamente, tutti i manoscritti sono stati presi come input dal sistema che ha generato i risultati riportati nella tabella 12.

TestGraphia	Paramether 1	Paramether 2	Paramether 3	Paramether 4	Paramether 8	Paramether 9
2,1	0	0	1	1	3	1
2,2	1	0	0	5	4	0
2,3	0	0	1	5	3	0
2,4	0	0	4	5	2	0
2,5	1	0	0	0	5	2
2,7	2	0	3	3	5	2
2,8*	5	0	2	3	4	2
3,1*	3	0	2	4	5	1
3,3	2	0	2	4	4	1
3,4	3	0	1	2	5	1
3,5*	3	1	2	5	5	2
3,6	3	0	1	5	5	0
3,7	4	0	2	5	4	0
3,8	2	0	0	5	4	1
4,1	1	0	0	1	5	2
5,3	3	0	1	0	5	0

Tabella 12: I parametri 1, 2, 3, 4, 8 e 9 sono stati identificati da TestGraphia per ogni bambino. Il primo numero della prima colonna indica la classe, mentre l'asterisco indica che il bambino è affetto da DSA.

Dopo la raccolta, per ogni bambino, sono state calcolate le differenze per ogni parametro. Per raggiungere questo obiettivo, abbiamo creato una tabella di sottrazione, campo per campo, dalla tabella 11 alla tabella 12. Se consideriamo le tabelle come matrici di interi, abbiamo che $S = A - B$. La tabella 13 mostra i risultati della matrice S.

Subtraction	Paramether 1	Paramether 2	Paramether 3	Paramether 4	Paramether 8	Paramether 9
2,1	0	0	0	0	0	0
2,2	0	0	0	0	0	-1
2,3	0	0	1	0	0	0
2,4	-1	0	1	0	0	0
2,5	-1	0	0	0	0	0
2,7	0	0	0	-1	0	-1
2,8	1	0	0	0	0	0
3,1	1	0	0	-1	0	0
3,3	0	0	0	0	0	0
3,4	0	0	0	0	0	0
3,5	0	0	1	0	0	0
3,6	1	0	0	1	0	0
3,7	1	0	0	2	0	0
3,8	0	0	0	0	0	0
4,1	-1	0	0	-1	0	-1
5,3	1	0	0	0	0	0

Tabella 13: Sottrazione tra i valori della matrice 11 e quelli della matrice 12.

Per identificare la correlazione è necessario identificare due indici: il primo indica la percentuale di parametri corrispondenti (N_{COR}) e il secondo lo scarto medio tra il valore

calcolato dal sistema e il valore attuale (P_{ACT}). Identificando con N il numero di campioni, P il numero di parametri da calcolare, $m_{i,j}$ il valore della riga i e della colonna j della matrice nella tabella di sottrazione, e sapendo che $N = 16$, $P=9$, il primo indice può essere calcolato in questo modo:

$$\forall i \in [1, \dots, N] \forall j \in [1, \dots, P]$$

$$s_{ij} = \begin{cases} 1 & \text{se } m_{i,j} \neq 0 \\ m_{i,j} & \text{else} \end{cases}$$

$$N_{COR} = \left(1 - \frac{1}{NP} \sum_{i=1}^N \sum_{j=1}^P s_{ij} \right) * 100$$

Il secondo indice, P_{ACT} , non conta quanti valori calcolati dal sistema sono diversi dall'originale, ma lo scarto medio.

$$P_{ACT} = \frac{\sum_{i=1}^N \sum_{j=1}^P \frac{|m_{ij}|}{\square}}{\sum_{i=1}^N \sum_{j=1}^P s_{ij}}$$

$$\text{se e solo se } \sum_{i=1}^N \sum_{j=1}^P s_{ij} \neq 0$$

Nel nostro caso, $N_{COR} = 80\%$ e $P_{ACT} = 1.05$.

IV.III Validazione dello specialista

Dopo la verifica con i nostri valori, è stato chiesto ad uno specialista di inserire manualmente i risultati di alcuni manoscritti. Per esigenze di tempo, il numero di test doveva essere fatto su un campione ridotto, in particolare sono stati scelti cinque manoscritti. I risultati sono stati raccolti nella tabella 14.

Expert	Paramether 1	Paramether 2	Paramether 3	Paramether 4	Paramether 8	Paramether 9
2,7	1	1	2	3	5	2
2,8	5	0	1	3	4	1
3,3	2	0	3	4	4	1
3,4	3	0	3	1	5	0
3,5	3	5	3	5	5	2

Tabella 14: I parametri 1, 2, 3, 4, 8 e 9 sono stati identificati manualmente per ogni bambino da uno specialista.

Analogamente a quanto fatto precedentemente, dopo la raccolta, per ogni bambino, sono state calcolate le differenze per ogni parametro. Per raggiungere questo obiettivo, abbiamo

creato una tabella di sottrazione, campo per campo, dalla tabella 14 alla tabella 12. Se consideriamo le tabelle come matrici di interi, abbiamo che $S' = A - B$. La tabella 15 mostra i risultati della matrice S' .

Subtraction	Paramether 1	Paramether 2	Paramether 3	Paramether 4	Paramether 8	Paramether 9
2,7	-1	1	-1	0	0	0
2,8	0	0	-1	0	0	-1
3,3	0	0	1	0	0	0
3,4	0	0	2	-1	0	-1
3,5	0	0	1	0	0	0

Tabella 15: Sottrazione tra i valori della matrice 14 e quelli della matrice 12.

Visto il numero ridotto di campioni, è difficile stabilire quanto siano concordi i risultati anche se i due indici possono essere ricalcolati. In questo caso $N_{COR} = 60\%$ e $P_{ACT} = 1.1$.

V. Ringraziamenti

Gli autori ringraziano la dottoressa Mara Bovino per il supporto in alcune decisioni riguardo la scelta dell'header da includere nel testo originale e l'interazione coi bambini. È stato inoltre fondamentale avere un riscontro per la validazione del sistema grazie ai suoi risultati forniti da alcuni manoscritti.

VI. Sviluppi futuri

Si considerano qui quelli che sono i miglioramenti futuri, che in realtà sono già in corso. Un'applicazione web è necessaria per garantire la diffusione del servizio. Esso è anche l'unico modo per poter raccogliere una grandissima quantità di campioni da esaminare riuscendo quindi a poter dare un nuovo confronto tra le diverse regioni d'Italia. L'applicazione web fungerà anche da gestionale per gli specialisti. Essi potranno registrarsi, inserire i loro pazienti e monitorare i loro risultati nel tempo. I pazienti potranno invece monitorare i loro progressi. Avere un gestionale in ambito medico è di grandissimo aiuto in quanto si potrebbe anche tener nota dei metodi terapeutici con i loro relativi risultati. Si potrà evitare di ripetere gli stessi errori ed avere una metodologia più centrata sul singolo paziente. Sarebbe anche auspicabile una certa interoperabilità con altri sistemi esterni che forniscono informazioni su tutta la condizione clinica del bambino. È fondamentale, inoltre, garantire un'ottima UI sia per specialisti che per pazienti in quanto bisogna cercare di rendere il più veloce possibile le operazioni di diagnosi e monitoraggio. Tutto ciò ha come punto chiave il cercare di diagnosticare problemi in anticipo o comunque in poco tempo. Questa versione standalone, in ogni caso, ha già adempiuto al compito di ridurre in maniera drastica il carico del lavoro del singolo specialista; si ritiene che successivi miglioramenti porteranno risultati ancora più soddisfacenti.

VII. Conclusioni

Il punto chiave di questo studio è quello di facilitare il compito degli esperti, riducendo il carico di lavoro e consentire una diffusione a largo raggio, rivelando eventualmente i primi sintomi. L'obiettivo è quello di essere in grado di esaminare più bambini in meno tempo con l'ovvia conseguenza di essere in grado di diagnosticare eventuali problemi in anticipo. Questo sistema permette anche una relazione esperto-paziente a distanza.

Il supporto fornito da questo sistema è significativo e, inoltre, libera l'esperto dall'uso della carta millimetrata mostrata in Figura 2, che richiede uno sforzo notevole e una particolare precisione.

Sono stati utilizzati algoritmi di analisi documentale, adattati a questo specifico obiettivo, come l'algoritmo di scrittura obliqua e l'algoritmo del margine sinistro non allineato.

VIII. Riferimenti

- [1] R. Ptucha, F. Petroski, S. Pillai, F. Brockler, V. Singh, and P. Hutkowski, "Intelligent character recognition using fully convolutional neural networks," *Pattern Recognit.*, vol. 88, pp. 604–613, 2019.
- [2] S. Rosenblum, P. L. Weiss, and S. Parush, "Handwriting evaluation for developmental dysgraphia :," no. 1993, pp. 1996–1997, 2004.
- [3] R. K. Deuel, "Development Dysgraphia and Motor Skills Disorders."
- [4] T. T. S. INGRAM, "Paediatric Aspects of Specific Developmental Dysphasia, Dyslexia and Dysgraphia."
- [5] E. Adi-Japha, "ADHD and Dysgraphia: Underlying Mechanisms."
- [6] J. Mortley, P. Enderby, and B. Petheram, "Using a computer to improve functional writing in a patient with severe dysgraphia."
- [7] J. Danna *et al.*, "Handwriting Movement Sonification for the Rehabilitation of Dysgraphia."
- [8] D. Caivano, F. Girardi, G. Dimauro, V. Bevilacqua, and V. Di Nicola, "Assessment of Speech Intelligibility in Parkinson's Disease Using a Speech-To-Text System," vol. 5, 2017.
- [9] C. Silveri, S. Misciagna, G. Leggio, and M. Molinari, "Spatial dysgraphia and cerebellar lesion."
- [10] "Dsa, Miur fotografa boom delle diagnosi: 67mila in più in un anno." [Online]. Available: <https://www.aiditalia.org/it/news-ed-eventi/news/quanti-sono-studenti-con-dsa-italia-dati-a-confronto>.
- [11] "Studenti con DSA in Italia: i dati MIUR per l'AS 2016/2017." [Online]. Available: <https://www.aiditalia.org/it/news-ed-eventi/news/analisi-e-commento-dati-miur-studenti-con-dsa-as-2016-2017>.
- [12] C. Di Brina and G. Rossini, "BHK: Scala sintetica per la valutazione della scrittura in età evolutiva."
- [13] P. Tressoldi, "I disturbi della scrittura."
- [14] U. Pal and S. Datta, "Segmentation of Bangla Unconstrained Handwritten Text," no. Icdar, pp. 3–7, 2003.
- [15] A. Nicolaou and G. Basilis, "Handwritten Text Line Segmentation by Shredding Text into its Lines."
- [16] Y. Li, Y. Zheng, D. Doermann, and S. Jaeger, "Script-Independent Text Line Segmentation in Freestyle Handwritten Documents."
- [17] Y. Cao, S. Wang, and H. Li, "Skew detection and correction in document images based on straight-line fitting."
- [18] B. G. Gatos, N. E. Papamarkos, and C. C. Chamzas, "Skew detection and text line position determination in digitized documents."
- [19] Z. Shi and V. Govindaraju, "Skew Detection for Complex Document Images Using Fuzzy Runlength."
- [20] Y. Ishitani, "Document skew detection based on local region complexity."