

# US Patent & Trademark Office

## Patent Public Search | Text View

---

United States Patent	12388937
Kind Code	B2
Date of Patent	August 12, 2025
Inventor(s)	D'Armancourt; Sebastien et al.

---

### Methods, apparatuses, and systems for detecting printing defects and contaminated components of a printer

---

#### Abstract

A method for printing defect detection includes processing and analyzing a difference image obtained by comparing an image scanned with a verifier to a reference image. The detected defects are grouped, and the grouping is refined. Confidence level values are then assigned to the refined groups, and analysis is performed to determine if one or more servicing actions should be taken.

---

**Inventors:** D'Armancourt; Sebastien (Singapore, SG), Celinder; Thomas Axel Jonas (Singapore, SG), Ackley; H Sprague (Seattle, WA), Aspenns; Glenn David (Cincinnati, OH)

**Applicant:** Hand Held Products, Inc. (Charlotte, NC)

**Family ID:** 65009608

**Assignee:** Hand Held Products, Inc. (Charlotte, NC)

**Appl. No.:** 18/423105

**Filed:** January 25, 2024

#### Prior Publication Data

Document Identifier	Publication Date
US 20240163385 A1	May. 16, 2024

#### Related U.S. Application Data

continuation parent-doc US 18146201 20221223 US 11943406 child-doc US 18423105  
continuation parent-doc US 17301259 20210330 US 11570321 20230131 child-doc US 18146201  
continuation parent-doc US 16790417 20200213 US 10999460 20210504 child-doc US 17301259  
continuation parent-doc US 16240140 20190104 US 10834283 20201110 child-doc US 16790417  
us-provisional-application US 62614089 20180105

---

## Publication Classification

**Int. Cl.:** **H04N1/00** (20060101); **G06K5/02** (20060101); **G06T7/00** (20170101); **G06T7/254** (20170101)

**U.S. Cl.:**

**CPC** **H04N1/00737** (20130101); **G06K5/02** (20130101); **G06T7/0002** (20130101); **G06T7/001** (20130101); **G06T7/254** (20170101); **H04N1/00005** (20130101); **H04N1/00015** (20130101); **H04N1/00029** (20130101); **H04N1/00037** (20130101); **H04N1/00039** (20130101); **H04N1/00045** (20130101); **H04N1/00082** (20130101); **G06T2207/30144** (20130101)

## Field of Classification Search

**CPC:** H04N (1/00737); H04N (1/00005); H04N (1/00015); H04N (1/00029); H04N (1/00037); H04N (1/00039); H04N (1/00045); H04N (1/00082); G06K (5/02); G06T (7/0002); G06T (7/001); G06T (7/254); G06T (2207/30144)

**USPC:** 358/504

---

## References Cited

### U.S. PATENT DOCUMENTS

Patent No.	Issued Date	Patentee Name	U.S. Cl.	CPC
4699531	12/1986	Ulinski et al.	N/A	N/A
5051567	12/1990	Tedesco	N/A	N/A
5218190	12/1992	Hardesty et al.	N/A	N/A
5272322	12/1992	Nishida et al.	N/A	N/A
5318938	12/1993	Hampl et al.	N/A	N/A
5488223	12/1995	Austin et al.	N/A	N/A
5488233	12/1995	Ishikawa et al.	N/A	N/A
5521368	12/1995	Adachi	N/A	N/A
5564841	12/1995	Austin et al.	N/A	N/A
5761336	12/1997	Xu et al.	N/A	N/A
5845008	12/1997	Katoh et al.	N/A	N/A
5914474	12/1998	Spitz	N/A	N/A
6036091	12/1999	Spitz	N/A	N/A
6042279	12/1999	Ackley	N/A	N/A
6323949	12/2000	Lading et al.	N/A	N/A
6511141	12/2002	Hasegawa et al.	N/A	N/A
6535299	12/2002	Scherz	N/A	N/A
6567530	12/2002	Keronen et al.	N/A	N/A
6741727	12/2003	Hirasawa	N/A	N/A
6763148	12/2003	Sternberg et al.	N/A	N/A
6832725	12/2003	Gardiner et al.	N/A	N/A
6997627	12/2005	Chiu	N/A	N/A
7128266	12/2005	Zhu et al.	N/A	N/A
7159783	12/2006	Walczyk et al.	N/A	N/A

7413127	12/2007	Ehrhart et al.	N/A	N/A
7440123	12/2007	Chodagiri et al.	N/A	N/A
7471331	12/2007	Kaneda	N/A	N/A
7519222	12/2008	Kisilev et al.	N/A	N/A
7570788	12/2008	Tsukamoto et al.	N/A	N/A
7600687	12/2008	Biss et al.	N/A	N/A
7726575	12/2009	Wang et al.	N/A	N/A
7869112	12/2010	Borchers et al.	N/A	N/A
7877004	12/2010	Maruyama et al.	N/A	N/A
7920283	12/2010	Shimazaki	N/A	N/A
7936365	12/2010	Jeong	N/A	N/A
8189229	12/2011	Mori	N/A	N/A
8269836	12/2011	Zandifar et al.	N/A	N/A
8294945	12/2011	Natori	N/A	N/A
8294969	12/2011	Plesko	N/A	N/A
8317105	12/2011	Kotlarsky et al.	N/A	N/A
8322622	12/2011	Liu	N/A	N/A
8355058	12/2012	Shirai	N/A	N/A
8366005	12/2012	Kotlarsky et al.	N/A	N/A
8371507	12/2012	Haggerty et al.	N/A	N/A
8376233	12/2012	Horn et al.	N/A	N/A
8381979	12/2012	Franz	N/A	N/A
8390909	12/2012	Plesko	N/A	N/A
8408464	12/2012	Zhu et al.	N/A	N/A
8408468	12/2012	Van et al.	N/A	N/A
8408469	12/2012	Good	N/A	N/A
8424768	12/2012	Rueblinger et al.	N/A	N/A
8448863	12/2012	Xian et al.	N/A	N/A
8457013	12/2012	Essinger et al.	N/A	N/A
8459557	12/2012	Havens et al.	N/A	N/A
8469272	12/2012	Kearney	N/A	N/A
8474712	12/2012	Kearney et al.	N/A	N/A
8479992	12/2012	Kotlarsky et al.	N/A	N/A
8482809	12/2012	Mikami	N/A	N/A
8488181	12/2012	Wu et al.	N/A	N/A
8490877	12/2012	Kearney	N/A	N/A
8517271	12/2012	Kotlarsky et al.	N/A	N/A
8523076	12/2012	Good	N/A	N/A
8528818	12/2012	Ehrhart et al.	N/A	N/A
8544737	12/2012	Gomez et al.	N/A	N/A
8548420	12/2012	Grunow et al.	N/A	N/A
8550335	12/2012	Samek et al.	N/A	N/A
8550354	12/2012	Gannon et al.	N/A	N/A
8550357	12/2012	Kearney	N/A	N/A
8556174	12/2012	Kosecki et al.	N/A	N/A
8556176	12/2012	Van et al.	N/A	N/A
8556177	12/2012	Hussey et al.	N/A	N/A
8559767	12/2012	Barber et al.	N/A	N/A
8561895	12/2012	Gomez et al.	N/A	N/A
8561903	12/2012	Sauerwein, Jr.	N/A	N/A

8561905	12/2012	Edmonds et al.	N/A	N/A
8565107	12/2012	Pease et al.	N/A	N/A
8571307	12/2012	Li et al.	N/A	N/A
8579200	12/2012	Samek et al.	N/A	N/A
8583924	12/2012	Caballero et al.	N/A	N/A
8584945	12/2012	Wang et al.	N/A	N/A
8587595	12/2012	Wang	N/A	N/A
8587697	12/2012	Hussey et al.	N/A	N/A
8588869	12/2012	Sauerwein et al.	N/A	N/A
8590789	12/2012	Nahill et al.	N/A	N/A
8596539	12/2012	Havens et al.	N/A	N/A
8596542	12/2012	Havens et al.	N/A	N/A
8596543	12/2012	Havens et al.	N/A	N/A
8599271	12/2012	Havens et al.	N/A	N/A
8599957	12/2012	Peake et al.	N/A	N/A
8600158	12/2012	Li et al.	N/A	N/A
8600167	12/2012	Showering	N/A	N/A
8602309	12/2012	Longacre et al.	N/A	N/A
8608053	12/2012	Meier et al.	N/A	N/A
8608071	12/2012	Liu et al.	N/A	N/A
8611309	12/2012	Wang et al.	N/A	N/A
8615487	12/2012	Gomez et al.	N/A	N/A
8621123	12/2012	Caballero	N/A	N/A
8622303	12/2013	Meier et al.	N/A	N/A
8628013	12/2013	Ding	N/A	N/A
8628015	12/2013	Wang et al.	N/A	N/A
8628016	12/2013	Winegar	N/A	N/A
8629926	12/2013	Wang	N/A	N/A
8630030	12/2013	Chung et al.	N/A	N/A
8630491	12/2013	Longacre et al.	N/A	N/A
8635309	12/2013	Berthiaume et al.	N/A	N/A
8636200	12/2013	Kearney	N/A	N/A
8636212	12/2013	Nahill et al.	N/A	N/A
8636215	12/2013	Ding et al.	N/A	N/A
8636224	12/2013	Wang	N/A	N/A
8638806	12/2013	Wang et al.	N/A	N/A
8640958	12/2013	Lu et al.	N/A	N/A
8640960	12/2013	Wang et al.	N/A	N/A
8643717	12/2013	Li et al.	N/A	N/A
8646692	12/2013	Meier et al.	N/A	N/A
8646694	12/2013	Wang et al.	N/A	N/A
8657200	12/2013	Ren et al.	N/A	N/A
8659397	12/2013	Vargo et al.	N/A	N/A
8668149	12/2013	Good	N/A	N/A
8675266	12/2013	Watts	N/A	N/A
8678285	12/2013	Kearney	N/A	N/A
8678286	12/2013	Smith et al.	N/A	N/A
8682077	12/2013	Longacre, Jr.	N/A	N/A
D702237	12/2013	Oberpriller et al.	N/A	N/A
8687282	12/2013	Feng et al.	N/A	N/A

8692927	12/2013	Pease et al.	N/A	N/A
8695880	12/2013	Bremer et al.	N/A	N/A
8698949	12/2013	Grunow et al.	N/A	N/A
8717494	12/2013	Gannon	N/A	N/A
8720783	12/2013	Biss et al.	N/A	N/A
8723804	12/2013	Fletcher et al.	N/A	N/A
8723904	12/2013	Marty et al.	N/A	N/A
8727223	12/2013	Wang	N/A	N/A
8736914	12/2013	French	N/A	N/A
8740082	12/2013	Wilz, Sr.	N/A	N/A
8740085	12/2013	Furlong et al.	N/A	N/A
8746563	12/2013	Hennick et al.	N/A	N/A
8750445	12/2013	Peake et al.	N/A	N/A
8752766	12/2013	Xian et al.	N/A	N/A
8756059	12/2013	Braho et al.	N/A	N/A
8757495	12/2013	Qu et al.	N/A	N/A
8760563	12/2013	Koziol et al.	N/A	N/A
8763909	12/2013	Reed et al.	N/A	N/A
8768102	12/2013	Ng et al.	N/A	N/A
8777108	12/2013	Coyle	N/A	N/A
8777109	12/2013	Oberpriller et al.	N/A	N/A
8779898	12/2013	Havens et al.	N/A	N/A
8781520	12/2013	Payne et al.	N/A	N/A
8783573	12/2013	Havens et al.	N/A	N/A
8789757	12/2013	Barten	N/A	N/A
8789758	12/2013	Hawley et al.	N/A	N/A
8789759	12/2013	Xian et al.	N/A	N/A
8794520	12/2013	Wang et al.	N/A	N/A
8794522	12/2013	Ehrhart	N/A	N/A
8794525	12/2013	Amundsen et al.	N/A	N/A
8794526	12/2013	Wang et al.	N/A	N/A
8798367	12/2013	Ellis	N/A	N/A
8807431	12/2013	Wang et al.	N/A	N/A
8807432	12/2013	Van et al.	N/A	N/A
8817327	12/2013	Kamijima	N/A	N/A
8820630	12/2013	Qu et al.	N/A	N/A
8822848	12/2013	Meagher	N/A	N/A
8824692	12/2013	Sheerin et al.	N/A	N/A
8824696	12/2013	Braho	N/A	N/A
8842849	12/2013	Wahl et al.	N/A	N/A
8844822	12/2013	Kotlarsky et al.	N/A	N/A
8844823	12/2013	Fritz et al.	N/A	N/A
8849019	12/2013	Li et al.	N/A	N/A
D716285	12/2013	Chaney et al.	N/A	N/A
8851383	12/2013	Yeakley et al.	N/A	N/A
8854633	12/2013	Laffargue et al.	N/A	N/A
8866963	12/2013	Grunow et al.	N/A	N/A
8868421	12/2013	Braho et al.	N/A	N/A
8868519	12/2013	Maloy et al.	N/A	N/A
8868802	12/2013	Barten	N/A	N/A

8868803	12/2013	Caballero	N/A	N/A
8870074	12/2013	Gannon	N/A	N/A
8879085	12/2013	Vandemark et al.	N/A	N/A
8879639	12/2013	Sauerwein, Jr.	N/A	N/A
8880426	12/2013	Smith	N/A	N/A
8881983	12/2013	Havens et al.	N/A	N/A
8881987	12/2013	Wang	N/A	N/A
8903172	12/2013	Smith	N/A	N/A
8908995	12/2013	Benos et al.	N/A	N/A
8910870	12/2013	Li et al.	N/A	N/A
8910875	12/2013	Ren et al.	N/A	N/A
8914290	12/2013	Hendrickson et al.	N/A	N/A
8914788	12/2013	Pettinelli et al.	N/A	N/A
8915439	12/2013	Feng et al.	N/A	N/A
8915444	12/2013	Havens et al.	N/A	N/A
8916789	12/2013	Woodburn	N/A	N/A
8918250	12/2013	Hollifield	N/A	N/A
8918564	12/2013	Caballero	N/A	N/A
8925818	12/2014	Kosecki et al.	N/A	N/A
8939374	12/2014	Jovanovski et al.	N/A	N/A
8942480	12/2014	Ellis	N/A	N/A
8944313	12/2014	Williams et al.	N/A	N/A
8944327	12/2014	Meier et al.	N/A	N/A
8944332	12/2014	Harding et al.	N/A	N/A
8950678	12/2014	Germaine et al.	N/A	N/A
D723560	12/2014	Zhou et al.	N/A	N/A
8967468	12/2014	Gomez et al.	N/A	N/A
8971346	12/2014	Sevier	N/A	N/A
8976030	12/2014	Cunningham et al.	N/A	N/A
8976368	12/2014	El et al.	N/A	N/A
8978981	12/2014	Guan	N/A	N/A
8978983	12/2014	Bremer et al.	N/A	N/A
8978984	12/2014	Hennick et al.	N/A	N/A
8985456	12/2014	Zhu et al.	N/A	N/A
8985457	12/2014	Soule et al.	N/A	N/A
8985459	12/2014	Kearney et al.	N/A	N/A
8985461	12/2014	Gelay et al.	N/A	N/A
8988578	12/2014	Showering	N/A	N/A
8988590	12/2014	Gillet et al.	N/A	N/A
8991704	12/2014	Hopper et al.	N/A	N/A
8996194	12/2014	Davis et al.	N/A	N/A
8996384	12/2014	Funyak et al.	N/A	N/A
9002641	12/2014	Showering	N/A	N/A
9007368	12/2014	Laffargue et al.	N/A	N/A
9010641	12/2014	Qu et al.	N/A	N/A
9015513	12/2014	Murawski et al.	N/A	N/A
9016576	12/2014	Brady et al.	N/A	N/A
D730357	12/2014	Fitch et al.	N/A	N/A
9022288	12/2014	Nahill et al.	N/A	N/A
9030964	12/2014	Essinger et al.	N/A	N/A

9033240	12/2014	Smith et al.	N/A	N/A
9033242	12/2014	Gillet et al.	N/A	N/A
9036037	12/2014	Rudin et al.	N/A	N/A
9036054	12/2014	Koziol et al.	N/A	N/A
9037344	12/2014	Chamberlin	N/A	N/A
9038911	12/2014	Xian et al.	N/A	N/A
9038915	12/2014	Smith	N/A	N/A
9041762	12/2014	Bai et al.	N/A	N/A
D730901	12/2014	Oberpriller et al.	N/A	N/A
D730902	12/2014	Fitch et al.	N/A	N/A
D733112	12/2014	Chaney et al.	N/A	N/A
9047098	12/2014	Barten	N/A	N/A
9047359	12/2014	Caballero et al.	N/A	N/A
9047420	12/2014	Caballero	N/A	N/A
9047525	12/2014	Barber et al.	N/A	N/A
9047531	12/2014	Showering et al.	N/A	N/A
9053055	12/2014	Caballero	N/A	N/A
9053378	12/2014	Hou et al.	N/A	N/A
9053380	12/2014	Xian et al.	N/A	N/A
9058526	12/2014	Powilleit	N/A	N/A
9064165	12/2014	Havens et al.	N/A	N/A
9064167	12/2014	Xian et al.	N/A	N/A
9064168	12/2014	Todeschini et al.	N/A	N/A
9064254	12/2014	Todeschini et al.	N/A	N/A
9066032	12/2014	Wang	N/A	N/A
9070032	12/2014	Corcoran	N/A	N/A
D734339	12/2014	Zhou et al.	N/A	N/A
D734751	12/2014	Oberpriller et al.	N/A	N/A
9079423	12/2014	Bouverie et al.	N/A	N/A
9093141	12/2014	Liu	N/A	N/A
9104934	12/2014	Li et al.	N/A	N/A
9111166	12/2014	Cunningham, IV	N/A	N/A
9137009	12/2014	Gardiner	N/A	N/A
9158000	12/2014	Sauerwein, Jr.	N/A	N/A
9158340	12/2014	Reed et al.	N/A	N/A
9158953	12/2014	Gillet et al.	N/A	N/A
9159059	12/2014	Daddabbo et al.	N/A	N/A
9171539	12/2014	Funyak et al.	N/A	N/A
9171543	12/2014	Emerick et al.	N/A	N/A
9174457	12/2014	Aihara et al.	N/A	N/A
9183425	12/2014	Wang	N/A	N/A
9189669	12/2014	Zhu et al.	N/A	N/A
9219836	12/2014	Bouverie et al.	N/A	N/A
9224022	12/2014	Ackley et al.	N/A	N/A
9224027	12/2014	Van et al.	N/A	N/A
D747321	12/2015	London et al.	N/A	N/A
9230140	12/2015	Ackley	N/A	N/A
9245492	12/2015	Ackley et al.	N/A	N/A
9248640	12/2015	Heng	N/A	N/A
9250712	12/2015	Todeschini	N/A	N/A

9258033	12/2015	Showering	N/A	N/A
9261398	12/2015	Amundsen et al.	N/A	N/A
9262633	12/2015	Todeschini et al.	N/A	N/A
9262662	12/2015	Chen et al.	N/A	N/A
9262664	12/2015	Soule et al.	N/A	N/A
9269036	12/2015	Bremer	N/A	N/A
9270782	12/2015	Hala et al.	N/A	N/A
9274806	12/2015	Barten	N/A	N/A
9274812	12/2015	Doren et al.	N/A	N/A
9275388	12/2015	Havens et al.	N/A	N/A
9277668	12/2015	Feng et al.	N/A	N/A
9280693	12/2015	Feng et al.	N/A	N/A
9282501	12/2015	Wang et al.	N/A	N/A
9292969	12/2015	Laffargue et al.	N/A	N/A
9298667	12/2015	Caballero	N/A	N/A
9298964	12/2015	Li et al.	N/A	N/A
9301427	12/2015	Feng et al.	N/A	N/A
D754205	12/2015	Nguyen et al.	N/A	N/A
D754206	12/2015	Nguyen et al.	N/A	N/A
9310609	12/2015	Rueblinger et al.	N/A	N/A
9317037	12/2015	Byford et al.	N/A	N/A
9319548	12/2015	Showering et al.	N/A	N/A
D757009	12/2015	Oberpriller et al.	N/A	N/A
9342724	12/2015	McCloskey et al.	N/A	N/A
9342827	12/2015	Smith	N/A	N/A
9355294	12/2015	Smith et al.	N/A	N/A
9361536	12/2015	Howe et al.	N/A	N/A
9361882	12/2015	Ressler et al.	N/A	N/A
9365381	12/2015	Colonel et al.	N/A	N/A
9367722	12/2015	Xian et al.	N/A	N/A
9373018	12/2015	Colavito et al.	N/A	N/A
9375945	12/2015	Bowles	N/A	N/A
D760719	12/2015	Zhou et al.	N/A	N/A
9390596	12/2015	Todeschini	N/A	N/A
9396375	12/2015	Qu et al.	N/A	N/A
9398008	12/2015	Todeschini et al.	N/A	N/A
D762604	12/2015	Fitch et al.	N/A	N/A
D762647	12/2015	Fitch et al.	N/A	N/A
9405011	12/2015	Showering	N/A	N/A
9407840	12/2015	Wang	N/A	N/A
9411386	12/2015	Sauerwein, Jr.	N/A	N/A
9412242	12/2015	Van et al.	N/A	N/A
9418252	12/2015	Nahill et al.	N/A	N/A
9418269	12/2015	Havens et al.	N/A	N/A
9418270	12/2015	Van Volkinburg et al.	N/A	N/A
9423318	12/2015	Liu et al.	N/A	N/A
D766244	12/2015	Zhou et al.	N/A	N/A
9443123	12/2015	Hejl	N/A	N/A
9443222	12/2015	Singel et al.	N/A	N/A
9448610	12/2015	Davis et al.	N/A	N/A



9454689	12/2015	McCloskey et al.	N/A	N/A
9478113	12/2015	Xie et al.	N/A	N/A
D771631	12/2015	Fitch et al.	N/A	N/A
9481186	12/2015	Bouverie et al.	N/A	N/A
9487113	12/2015	Schukalski	N/A	N/A
9488986	12/2015	Solanki	N/A	N/A
9489782	12/2015	Payne et al.	N/A	N/A
9490540	12/2015	Davies et al.	N/A	N/A
9491729	12/2015	Rautiola et al.	N/A	N/A
9497092	12/2015	Gomez et al.	N/A	N/A
9507974	12/2015	Todeschini	N/A	N/A
9519814	12/2015	Cudzilo	N/A	N/A
9521331	12/2015	Bessettes et al.	N/A	N/A
D777166	12/2016	Bidwell et al.	N/A	N/A
9557166	12/2016	Thuries et al.	N/A	N/A
9558386	12/2016	Yeakley	N/A	N/A
9564035	12/2016	Ackley et al.	N/A	N/A
9569837	12/2016	Madden et al.	N/A	N/A
9582696	12/2016	Barber et al.	N/A	N/A
9606581	12/2016	Howe et al.	N/A	N/A
D783601	12/2016	Schulte et al.	N/A	N/A
9616749	12/2016	Chamberlin	N/A	N/A
9618993	12/2016	Murawski et al.	N/A	N/A
D785617	12/2016	Bidwell et al.	N/A	N/A
D785636	12/2016	Oberpriller et al.	N/A	N/A
9641700	12/2016	Schumann et al.	N/A	N/A
9646189	12/2016	Lu et al.	N/A	N/A
9646200	12/2016	Archibald et al.	N/A	N/A
9652648	12/2016	Ackley et al.	N/A	N/A
9652653	12/2016	Todeschini et al.	N/A	N/A
9656487	12/2016	Ho et al.	N/A	N/A
9659183	12/2016	Zhu et al.	N/A	N/A
9659198	12/2016	Giordano et al.	N/A	N/A
9659670	12/2016	Choi et al.	N/A	N/A
9665757	12/2016	Feng et al.	N/A	N/A
D790505	12/2016	Vargo et al.	N/A	N/A
D790546	12/2016	Zhou et al.	N/A	N/A
D790553	12/2016	Fitch et al.	N/A	N/A
9680282	12/2016	Hanenbourg	N/A	N/A
9697401	12/2016	Feng et al.	N/A	N/A
9701140	12/2016	Alaganchetty et al.	N/A	N/A
9712758	12/2016	Noda	N/A	N/A
9715614	12/2016	Todeschini et al.	N/A	N/A
9734493	12/2016	Gomez et al.	N/A	N/A
9752864	12/2016	Laffargue et al.	N/A	N/A
9762793	12/2016	Ackley et al.	N/A	N/A
9767581	12/2016	Todeschini	N/A	N/A
9786101	12/2016	Ackley	N/A	N/A
9794392	12/2016	Hejl	N/A	N/A
9823059	12/2016	Li et al.	N/A	N/A

9826106	12/2016	Ackley	N/A	N/A
9852102	12/2016	Kohtz et al.	N/A	N/A
9857167	12/2017	Jovanovski et al.	N/A	N/A
9861182	12/2017	Oberpriller et al.	N/A	N/A
9891612	12/2017	Charpentier et al.	N/A	N/A
9892876	12/2017	Bandringa	N/A	N/A
9897434	12/2017	Ackley et al.	N/A	N/A
9898814	12/2017	Kitai et al.	N/A	N/A
9924006	12/2017	Schoon et al.	N/A	N/A
9930050	12/2017	Yeakley et al.	N/A	N/A
9954871	12/2017	Hussey et al.	N/A	N/A
9978088	12/2017	Pape	N/A	N/A
9984685	12/2017	Braho et al.	N/A	N/A
10007112	12/2017	Fitch et al.	N/A	N/A
10019334	12/2017	Caballero et al.	N/A	N/A
10021043	12/2017	Sevier	N/A	N/A
10038716	12/2017	Todeschini et al.	N/A	N/A
10060729	12/2017	Laffargue et al.	N/A	N/A
10066982	12/2017	Ackley et al.	N/A	N/A
10121466	12/2017	Pecorari	N/A	N/A
10139495	12/2017	Payne	N/A	N/A
10195880	12/2018	D'Armancourt	N/A	N/A
10269342	12/2018	Braho et al.	N/A	N/A
10427424	12/2018	Creencia et al.	N/A	N/A
10546160	12/2019	Sprague et al.	N/A	N/A
10552848	12/2019	Soborski	N/A	N/A
10621470	12/2019	Ackley	N/A	N/A
10628723	12/2019	D'Armancourt et al.	N/A	N/A
10672588	12/2019	Pathangi et al.	N/A	N/A
10795618	12/2019	Gutierrez et al.	N/A	N/A
10834283	12/2019	D'Armancourt et al.	N/A	N/A
10999460	12/2020	D'Armancourt et al.	N/A	N/A
11157217	12/2020	Gutierrez et al.	N/A	N/A
11210483	12/2020	Ackley et al.	N/A	N/A
2001/0016054	12/2000	Banker et al.	N/A	N/A
2001/0035971	12/2000	Koakutsu et al.	N/A	N/A
2002/0067925	12/2001	Yamano	N/A	N/A
2002/0181805	12/2001	Loeb et al.	N/A	N/A
2003/0102376	12/2002	Meier et al.	N/A	N/A
2003/0142985	12/2002	Sampath	399/9	G03G 15/5062
2004/0029068	12/2003	Sachdeva et al.	N/A	N/A
2004/0033098	12/2003	Chiu	N/A	N/A
2004/0036089	12/2003	Chen	N/A	N/A
2004/0057768	12/2003	Oshino et al.	N/A	N/A
2004/0120569	12/2003	Hung et al.	N/A	N/A
2004/0156630	12/2003	Tsukamoto et al.	N/A	N/A
2005/0105104	12/2004	Sakai et al.	N/A	N/A
2006/0012664	12/2005	Jeong	N/A	N/A
2006/0039690	12/2005	Steinberg et al.	N/A	N/A

2006/0110024	12/2002	Wakabayashi	N/A	N/A
2006/0197795	12/2005	Takatsuka et al.	N/A	N/A
2006/0269342	12/2005	Yoshida et al.	N/A	N/A
2007/0139703	12/2006	Shimamura et al.	N/A	N/A
2007/0146755	12/2006	Mindler et al.	N/A	N/A
2007/0195337	12/2006	Takayama et al.	N/A	N/A
2008/0040064	12/2007	Ishikawa	702/108	G01N 21/9501
2008/0144080	12/2007	Randt	N/A	N/A
2008/0185432	12/2007	Caballero et al.	N/A	N/A
2008/0218551	12/2007	Inaba	N/A	N/A
2009/0002749	12/2008	Koyano	N/A	N/A
2009/0058348	12/2008	Ryu	N/A	N/A
2009/0085952	12/2008	Yamazaki	N/A	N/A
2009/0087022	12/2008	Fukuda et al.	N/A	N/A
2009/0134221	12/2008	Zhu et al.	N/A	N/A
2009/0316161	12/2008	Yamaguchi et al.	N/A	N/A
2010/0165022	12/2009	Makuta et al.	N/A	N/A
2010/0177076	12/2009	Essinger et al.	N/A	N/A
2010/0177080	12/2009	Essinger et al.	N/A	N/A
2010/0177707	12/2009	Essinger et al.	N/A	N/A
2010/0177749	12/2009	Essinger et al.	N/A	N/A
2010/0182618	12/2009	Akiyama et al.	N/A	N/A
2010/0188714	12/2009	Yamakawa	N/A	N/A
2010/0265880	12/2009	Rautiola et al.	N/A	N/A
2011/0102850	12/2010	Watanabe	N/A	N/A
2011/0109918	12/2010	Conlon et al.	N/A	N/A
2011/0202554	12/2010	Powilleit et al.	N/A	N/A
2011/0286043	12/2010	Hagisawa et al.	N/A	N/A
2011/0292435	12/2010	Cok et al.	N/A	N/A
2012/0039384	12/2011	Reznik	N/A	N/A
2012/0085823	12/2011	Nakamura	N/A	N/A
2012/0111946	12/2011	Golant	N/A	N/A
2012/0168511	12/2011	Kotlarsky et al.	N/A	N/A
2012/0168512	12/2011	Kotlarsky et al.	N/A	N/A
2012/0182374	12/2011	Matsuda et al.	N/A	N/A
2012/0182571	12/2011	Wu et al.	N/A	N/A
2012/0193423	12/2011	Samek	N/A	N/A
2012/0203647	12/2011	Smith	N/A	N/A
2012/0223141	12/2011	Good et al.	N/A	N/A
2012/0228382	12/2011	Havens et al.	N/A	N/A
2012/0248188	12/2011	Kearney	N/A	N/A
2012/0263483	12/2011	Suzuki	N/A	N/A
2012/0330447	12/2011	Gerlach et al.	N/A	N/A
2013/0038670	12/2012	Chen	N/A	N/A
2013/0043312	12/2012	Van Horn	N/A	N/A
2013/0082104	12/2012	Kearney et al.	N/A	N/A
2013/0148987	12/2012	Arakawa	N/A	N/A
2013/0153661	12/2012	Yamaguchi	N/A	N/A
2013/0175341	12/2012	Kearney et al.	N/A	N/A

2013/0175343	12/2012	Good	N/A	N/A
2013/0250369	12/2012	Kitai et al.	N/A	N/A
2013/0250370	12/2012	Kojima et al.	N/A	N/A
2013/0257744	12/2012	Daghigh et al.	N/A	N/A
2013/0257759	12/2012	Daghigh	N/A	N/A
2013/0258368	12/2012	Shigemoto et al.	N/A	N/A
2013/0259301	12/2012	Chen et al.	N/A	N/A
2013/0270346	12/2012	Xian et al.	N/A	N/A
2013/0287258	12/2012	Kearney	N/A	N/A
2013/0292475	12/2012	Kotlarsky et al.	N/A	N/A
2013/0292477	12/2012	Hennick et al.	N/A	N/A
2013/0293539	12/2012	Hunt et al.	N/A	N/A
2013/0293540	12/2012	Laffargue et al.	N/A	N/A
2013/0306728	12/2012	Thuries et al.	N/A	N/A
2013/0306731	12/2012	Pedrao	N/A	N/A
2013/0307964	12/2012	Bremer et al.	N/A	N/A
2013/0308625	12/2012	Park et al.	N/A	N/A
2013/0313324	12/2012	Koziol et al.	N/A	N/A
2013/0313325	12/2012	Wilz et al.	N/A	N/A
2013/0322701	12/2012	Szymanski	N/A	N/A
2013/0332996	12/2012	Fiala et al.	N/A	N/A
2013/0335759	12/2012	Hirose	N/A	N/A
2013/0342717	12/2012	Havens et al.	N/A	N/A
2014/0001267	12/2013	Giordano et al.	N/A	N/A
2014/0002842	12/2013	Ito	N/A	N/A
2014/0008439	12/2013	Wang	N/A	N/A
2014/0009529	12/2013	Teshigawara et al.	N/A	N/A
2014/0025584	12/2013	Liu et al.	N/A	N/A
2014/0034734	12/2013	Sauerwein, Jr.	N/A	N/A
2014/0036848	12/2013	Pease et al.	N/A	N/A
2014/0039693	12/2013	Havens et al.	N/A	N/A
2014/0042814	12/2013	Kather et al.	N/A	N/A
2014/0049120	12/2013	Kohtz et al.	N/A	N/A
2014/0049635	12/2013	Laffargue et al.	N/A	N/A
2014/0061306	12/2013	Wu et al.	N/A	N/A
2014/0063289	12/2013	Hussey et al.	N/A	N/A
2014/0066136	12/2013	Sauerwein et al.	N/A	N/A
2014/0067692	12/2013	Ye et al.	N/A	N/A
2014/0070005	12/2013	Nahill et al.	N/A	N/A
2014/0071840	12/2013	Venancio	N/A	N/A
2014/0074746	12/2013	Wang	N/A	N/A
2014/0076974	12/2013	Havens et al.	N/A	N/A
2014/0078341	12/2013	Havens et al.	N/A	N/A
2014/0078345	12/2013	Showering	N/A	N/A
2014/0079292	12/2013	Kaneko et al.	N/A	N/A
2014/0097249	12/2013	Gomez et al.	N/A	N/A
2014/0098792	12/2013	Wang et al.	N/A	N/A
2014/0100813	12/2013	Showering	N/A	N/A
2014/0103115	12/2013	Meier et al.	N/A	N/A
2014/0104413	12/2013	McCloskey et al.	N/A	N/A

2014/0104414	12/2013	McCloskey et al.	N/A	N/A
2014/0104416	12/2013	Giordano et al.	N/A	N/A
2014/0104451	12/2013	Todeschini et al.	N/A	N/A
2014/0106594	12/2013	Skvoretz	N/A	N/A
2014/0106725	12/2013	Sauerwein, Jr.	N/A	N/A
2014/0108010	12/2013	Maltseff et al.	N/A	N/A
2014/0108402	12/2013	Gomez et al.	N/A	N/A
2014/0110485	12/2013	Toa et al.	N/A	N/A
2014/0114530	12/2013	Fitch et al.	N/A	N/A
2014/0124577	12/2013	Wang et al.	N/A	N/A
2014/0124579	12/2013	Ding	N/A	N/A
2014/0125842	12/2013	Winegar	N/A	N/A
2014/0125853	12/2013	Wang	N/A	N/A
2014/0125999	12/2013	Longacre et al.	N/A	N/A
2014/0129378	12/2013	Richardson	N/A	N/A
2014/0131438	12/2013	Kearney	N/A	N/A
2014/0131441	12/2013	Nahill et al.	N/A	N/A
2014/0131443	12/2013	Smith	N/A	N/A
2014/0131444	12/2013	Wang	N/A	N/A
2014/0131445	12/2013	Ding et al.	N/A	N/A
2014/0131448	12/2013	Xian et al.	N/A	N/A
2014/0133379	12/2013	Wang et al.	N/A	N/A
2014/0136208	12/2013	Maltseff et al.	N/A	N/A
2014/0140585	12/2013	Wang	N/A	N/A
2014/0151453	12/2013	Meier et al.	N/A	N/A
2014/0152882	12/2013	Samek et al.	N/A	N/A
2014/0158770	12/2013	Sevier et al.	N/A	N/A
2014/0159869	12/2013	Zumsteg et al.	N/A	N/A
2014/0166755	12/2013	Liu et al.	N/A	N/A
2014/0166759	12/2013	Liu et al.	N/A	N/A
2014/0168709	12/2013	Tokumaru	N/A	N/A
2014/0168787	12/2013	Wang et al.	N/A	N/A
2014/0175165	12/2013	Havens et al.	N/A	N/A
2014/0175172	12/2013	Jovanovski et al.	N/A	N/A
2014/0191644	12/2013	Chaney	N/A	N/A
2014/0191684	12/2013	Valois	N/A	N/A
2014/0191913	12/2013	Ge et al.	N/A	N/A
2014/0197238	12/2013	Liu et al.	N/A	N/A
2014/0197239	12/2013	Havens et al.	N/A	N/A
2014/0197304	12/2013	Feng et al.	N/A	N/A
2014/0214631	12/2013	Hansen	N/A	N/A
2014/0217166	12/2013	Berthiaume et al.	N/A	N/A
2014/0217180	12/2013	Liu	N/A	N/A
2014/0231500	12/2013	Ehrhart et al.	N/A	N/A
2014/0232930	12/2013	Anderson	N/A	N/A
2014/0247315	12/2013	Marty et al.	N/A	N/A
2014/0263493	12/2013	Amurgis et al.	N/A	N/A
2014/0263645	12/2013	Smith et al.	N/A	N/A
2014/0267609	12/2013	Laffargue	N/A	N/A
2014/0270196	12/2013	Braho et al.	N/A	N/A

2014/0270229	12/2013	Braho	N/A	N/A
2014/0278387	12/2013	Digregorio	N/A	N/A
2014/0278391	12/2013	Braho et al.	N/A	N/A
2014/0282210	12/2013	Bianconi	N/A	N/A
2014/0284384	12/2013	Lu et al.	N/A	N/A
2014/0288933	12/2013	Braho et al.	N/A	N/A
2014/0291401	12/2013	Nakamura	N/A	N/A
2014/0297058	12/2013	Barker et al.	N/A	N/A
2014/0299665	12/2013	Barber et al.	N/A	N/A
2014/0312121	12/2013	Lu et al.	N/A	N/A
2014/0319220	12/2013	Coyle	N/A	N/A
2014/0319221	12/2013	Oberpriller et al.	N/A	N/A
2014/0326787	12/2013	Barten	N/A	N/A
2014/0332590	12/2013	Wang et al.	N/A	N/A
2014/0344943	12/2013	Todeschini et al.	N/A	N/A
2014/0346233	12/2013	Liu et al.	N/A	N/A
2014/0351317	12/2013	Smith et al.	N/A	N/A
2014/0353373	12/2013	Van et al.	N/A	N/A
2014/0361073	12/2013	Qu et al.	N/A	N/A
2014/0361082	12/2013	Xian et al.	N/A	N/A
2014/0362184	12/2013	Jovanovski et al.	N/A	N/A
2014/0363015	12/2013	Braho	N/A	N/A
2014/0369511	12/2013	Sheerin et al.	N/A	N/A
2014/0374483	12/2013	Lu	N/A	N/A
2014/0374485	12/2013	Xian et al.	N/A	N/A
2015/0001301	12/2014	Ouyang	N/A	N/A
2015/0001304	12/2014	Todeschini	N/A	N/A
2015/0003673	12/2014	Fletcher	N/A	N/A
2015/0009338	12/2014	Laffargue et al.	N/A	N/A
2015/0009610	12/2014	London et al.	N/A	N/A
2015/0014416	12/2014	Kotlarsky et al.	N/A	N/A
2015/0021397	12/2014	Rueblinger et al.	N/A	N/A
2015/0028102	12/2014	Ren et al.	N/A	N/A
2015/0028103	12/2014	Jiang	N/A	N/A
2015/0028104	12/2014	Ma et al.	N/A	N/A
2015/0029002	12/2014	Yeakley et al.	N/A	N/A
2015/0032709	12/2014	Maloy et al.	N/A	N/A
2015/0039309	12/2014	Braho et al.	N/A	N/A
2015/0039878	12/2014	Barten	N/A	N/A
2015/0040378	12/2014	Saber et al.	N/A	N/A
2015/0048168	12/2014	Fritz et al.	N/A	N/A
2015/0049347	12/2014	Laffargue et al.	N/A	N/A
2015/0051992	12/2014	Smith	N/A	N/A
2015/0053766	12/2014	Havens et al.	N/A	N/A
2015/0053768	12/2014	Wang et al.	N/A	N/A
2015/0053769	12/2014	Thuries et al.	N/A	N/A
2015/0060544	12/2014	Feng et al.	N/A	N/A
2015/0062366	12/2014	Liu et al.	N/A	N/A
2015/0063215	12/2014	Wang	N/A	N/A
2015/0063676	12/2014	Lloyd et al.	N/A	N/A

2015/0063889	12/2014	Kojima	N/A	N/A
2015/0069130	12/2014	Gannon	N/A	N/A
2015/0071819	12/2014	Todeschini	N/A	N/A
2015/0078627	12/2014	Fukase	N/A	N/A
2015/0083800	12/2014	Li et al.	N/A	N/A
2015/0086114	12/2014	Todeschini	N/A	N/A
2015/0088522	12/2014	Hendrickson et al.	N/A	N/A
2015/0096872	12/2014	Woodburn	N/A	N/A
2015/0099557	12/2014	Pettinelli et al.	N/A	N/A
2015/0100196	12/2014	Hollifield	N/A	N/A
2015/0102109	12/2014	Huck	N/A	N/A
2015/0115035	12/2014	Meier et al.	N/A	N/A
2015/0127791	12/2014	Kosecki et al.	N/A	N/A
2015/0128116	12/2014	Chen et al.	N/A	N/A
2015/0129659	12/2014	Feng et al.	N/A	N/A
2015/0133047	12/2014	Smith et al.	N/A	N/A
2015/0134470	12/2014	Hejl et al.	N/A	N/A
2015/0136851	12/2014	Harding et al.	N/A	N/A
2015/0136854	12/2014	Lu et al.	N/A	N/A
2015/0142492	12/2014	Kumar	N/A	N/A
2015/0144692	12/2014	Hejl	N/A	N/A
2015/0144698	12/2014	Teng et al.	N/A	N/A
2015/0144701	12/2014	Xian et al.	N/A	N/A
2015/0149946	12/2014	Benos et al.	N/A	N/A
2015/0161429	12/2014	Xian	N/A	N/A
2015/0169925	12/2014	Chen et al.	N/A	N/A
2015/0169929	12/2014	Williams et al.	N/A	N/A
2015/0178523	12/2014	Gelay et al.	N/A	N/A
2015/0178534	12/2014	Jovanovski et al.	N/A	N/A
2015/0178535	12/2014	Bremer et al.	N/A	N/A
2015/0178536	12/2014	Hennick et al.	N/A	N/A
2015/0178537	12/2014	El et al.	N/A	N/A
2015/0181093	12/2014	Zhu et al.	N/A	N/A
2015/0181109	12/2014	Gillet et al.	N/A	N/A
2015/0193644	12/2014	Kearney et al.	N/A	N/A
2015/0220753	12/2014	Zhu et al.	N/A	N/A
2015/0221077	12/2014	Kawabata	382/141	G06T 7/337
2015/0281019	12/2014	Hashizume	N/A	N/A
2015/0312780	12/2014	Wang et al.	N/A	N/A
2015/0324623	12/2014	Powilleit	N/A	N/A
2015/0327012	12/2014	Bian et al.	N/A	N/A
2016/0025697	12/2015	Alt et al.	N/A	N/A
2016/0042241	12/2015	Todeschini	N/A	N/A
2016/0057230	12/2015	Todeschini et al.	N/A	N/A
2016/0062473	12/2015	Bouchat et al.	N/A	N/A
2016/0092805	12/2015	Geisler et al.	N/A	N/A
2016/0101936	12/2015	Chamberlin	N/A	N/A
2016/0104019	12/2015	Todeschini et al.	N/A	N/A
2016/0104274	12/2015	Jovanovski et al.	N/A	N/A
2016/0117627	12/2015	Raj et al.	N/A	N/A

2016/0125217	12/2015	Todeschini	N/A	N/A
2016/0125342	12/2015	Miller et al.	N/A	N/A
2016/0171597	12/2015	Todeschini	N/A	N/A
2016/0171666	12/2015	McCloskey	N/A	N/A
2016/0171775	12/2015	Todeschini et al.	N/A	N/A
2016/0171777	12/2015	Todeschini et al.	N/A	N/A
2016/0174674	12/2015	Oberpriller et al.	N/A	N/A
2016/0178479	12/2015	Goldsmith	N/A	N/A
2016/0178685	12/2015	Young et al.	N/A	N/A
2016/0178707	12/2015	Young et al.	N/A	N/A
2016/0179132	12/2015	Harr	N/A	N/A
2016/0179143	12/2015	Bidwell et al.	N/A	N/A
2016/0179368	12/2015	Roeder	N/A	N/A
2016/0179378	12/2015	Kent et al.	N/A	N/A
2016/0180130	12/2015	Bremer	N/A	N/A
2016/0180133	12/2015	Oberpriller et al.	N/A	N/A
2016/0180136	12/2015	Meier et al.	N/A	N/A
2016/0180594	12/2015	Todeschini	N/A	N/A
2016/0180663	12/2015	McMahan et al.	N/A	N/A
2016/0180713	12/2015	Bernhardt et al.	N/A	N/A
2016/0185136	12/2015	Ng et al.	N/A	N/A
2016/0185291	12/2015	Chamberlin	N/A	N/A
2016/0186926	12/2015	Oberpriller et al.	N/A	N/A
2016/0188861	12/2015	Todeschini	N/A	N/A
2016/0188939	12/2015	Sailors et al.	N/A	N/A
2016/0188940	12/2015	Lu et al.	N/A	N/A
2016/0188941	12/2015	Todeschini et al.	N/A	N/A
2016/0188942	12/2015	Good et al.	N/A	N/A
2016/0188943	12/2015	Franz	N/A	N/A
2016/0188944	12/2015	Wilz, Sr. et al.	N/A	N/A
2016/0189076	12/2015	Mellott et al.	N/A	N/A
2016/0189087	12/2015	Morton et al.	N/A	N/A
2016/0189088	12/2015	Pecorari et al.	N/A	N/A
2016/0189092	12/2015	George et al.	N/A	N/A
2016/0189284	12/2015	Mellott et al.	N/A	N/A
2016/0189288	12/2015	Todeschini et al.	N/A	N/A
2016/0189366	12/2015	Chamberlin et al.	N/A	N/A
2016/0189443	12/2015	Smith	N/A	N/A
2016/0189447	12/2015	Valenzuela	N/A	N/A
2016/0189489	12/2015	Au et al.	N/A	N/A
2016/0192051	12/2015	Dipiazza et al.	N/A	N/A
2016/0202951	12/2015	Pike et al.	N/A	N/A
2016/0202958	12/2015	Zabel et al.	N/A	N/A
2016/0202959	12/2015	Doubleday et al.	N/A	N/A
2016/0203021	12/2015	Pike et al.	N/A	N/A
2016/0203429	12/2015	Mellott et al.	N/A	N/A
2016/0203797	12/2015	Pike et al.	N/A	N/A
2016/0203820	12/2015	Zabel et al.	N/A	N/A
2016/0204623	12/2015	Haggerty et al.	N/A	N/A
2016/0204636	12/2015	Allen et al.	N/A	N/A



2016/0204638	12/2015	Miraglia et al.	N/A	N/A
2016/0255241	12/2015	Harashima et al.	N/A	N/A
2016/0282807	12/2015	Kinoshita et al.	N/A	N/A
2016/0292477	12/2015	Bidwell	N/A	N/A
2016/0314276	12/2015	Wilz et al.	N/A	N/A
2016/0314294	12/2015	Kubler et al.	N/A	N/A
2016/0316190	12/2015	McCloskey et al.	N/A	N/A
2016/0327614	12/2015	Young et al.	N/A	N/A
2016/0343163	12/2015	Venkatesha et al.	N/A	N/A
2016/0364914	12/2015	Todeschini	N/A	N/A
2016/0373847	12/2015	Vargo et al.	N/A	N/A
2016/0377414	12/2015	Thuries et al.	N/A	N/A
2017/0010141	12/2016	Ackley	N/A	N/A
2017/0010328	12/2016	Mullen et al.	N/A	N/A
2017/0010780	12/2016	Waldron, Jr. et al.	N/A	N/A
2017/0016714	12/2016	Laffargue et al.	N/A	N/A
2017/0018094	12/2016	Todeschini	N/A	N/A
2017/0046603	12/2016	Lee et al.	N/A	N/A
2017/0047864	12/2016	Stang et al.	N/A	N/A
2017/0053146	12/2016	Liu et al.	N/A	N/A
2017/0053147	12/2016	Germaine et al.	N/A	N/A
2017/0053647	12/2016	Nichols et al.	N/A	N/A
2017/0055606	12/2016	Xu et al.	N/A	N/A
2017/0060316	12/2016	Larson	N/A	N/A
2017/0060494	12/2016	Palmen et al.	N/A	N/A
2017/0061961	12/2016	Nichols et al.	N/A	N/A
2017/0064634	12/2016	Van Horn et al.	N/A	N/A
2017/0083730	12/2016	Feng et al.	N/A	N/A
2017/0083734	12/2016	Henning et al.	N/A	N/A
2017/0091502	12/2016	Furlong et al.	N/A	N/A
2017/0091706	12/2016	Lloyd et al.	N/A	N/A
2017/0091741	12/2016	Todeschini	N/A	N/A
2017/0091904	12/2016	Ventress, Jr.	N/A	N/A
2017/0092908	12/2016	Chaney	N/A	N/A
2017/0094238	12/2016	Germaine et al.	N/A	N/A
2017/0098947	12/2016	Wolski	N/A	N/A
2017/0100949	12/2016	Celinder et al.	N/A	N/A
2017/0108838	12/2016	Todeschini et al.	N/A	N/A
2017/0108895	12/2016	Chamberlin et al.	N/A	N/A
2017/0111523	12/2016	Ackley	N/A	N/A
2017/0118355	12/2016	Wong et al.	N/A	N/A
2017/0123598	12/2016	Phan et al.	N/A	N/A
2017/0124369	12/2016	Rueblinger et al.	N/A	N/A
2017/0124396	12/2016	Todeschini et al.	N/A	N/A
2017/0124687	12/2016	McCloskey et al.	N/A	N/A
2017/0126873	12/2016	McGary et al.	N/A	N/A
2017/0126904	12/2016	D'Armancourt et al.	N/A	N/A
2017/0139012	12/2016	Smith	N/A	N/A
2017/0140329	12/2016	Bernhardt et al.	N/A	N/A
2017/0140731	12/2016	Smith	N/A	N/A

2017/0147847	12/2016	Berggren et al.	N/A	N/A
2017/0150124	12/2016	Thuries	N/A	N/A
2017/0169198	12/2016	Nichols	N/A	N/A
2017/0171035	12/2016	Lu et al.	N/A	N/A
2017/0171703	12/2016	Maheswaranathan	N/A	N/A
2017/0171803	12/2016	Maheswaranathan	N/A	N/A
2017/0180359	12/2016	Wolski et al.	N/A	N/A
2017/0180577	12/2016	Nguon et al.	N/A	N/A
2017/0181299	12/2016	Shi et al.	N/A	N/A
2017/0182819	12/2016	Gonzalez et al.	N/A	N/A
2017/0190192	12/2016	Delario et al.	N/A	N/A
2017/0193432	12/2016	Bernhardt	N/A	N/A
2017/0193461	12/2016	Celinder et al.	N/A	N/A
2017/0193727	12/2016	Van Horn et al.	N/A	N/A
2017/0199266	12/2016	Rice et al.	N/A	N/A
2017/0200108	12/2016	Au et al.	N/A	N/A
2017/0200275	12/2016	McCloskey et al.	N/A	N/A
2017/0206643	12/2016	Weiss et al.	N/A	N/A
2017/0309011	12/2016	Hori et al.	N/A	N/A
2018/0007215	12/2017	Zakharov	N/A	N/A
2018/0124257	12/2017	Ueda	N/A	H04N 1/00079
2018/0131815	12/2017	Spivakovsky et al.	N/A	N/A
2018/0198937	12/2017	Yoshizawa	N/A	H04N 1/00702
2018/0227463	12/2017	Fukase	N/A	N/A
2018/0268534	12/2017	Kaneko	N/A	N/A
2019/0116275	12/2018	Edwards et al.	N/A	N/A
2019/0213369	12/2018	Ackley et al.	N/A	N/A
2019/0215410	12/2018	D'Armancourt et al.	N/A	N/A
2020/0082131	12/2019	Ackley et al.	N/A	N/A
2020/0118263	12/2019	Nogami	N/A	G06T 7/73
2020/0145546	12/2019	Alaganchetty et al.	N/A	N/A
2020/0234422	12/2019	Esumi et al.	N/A	N/A
2020/0401774	12/2019	Ackley	N/A	G06K 7/14
2021/0045268	12/2020	Easton et al.	N/A	N/A
2021/0197601	12/2020	Celinder et al.	N/A	N/A
2021/0218857	12/2020	D'Armancourt et al.	N/A	N/A
2021/0266422	12/2020	Kikuta	N/A	H04N 1/40006
2021/0295117	12/2020	Lim et al.	N/A	N/A
2021/0405938	12/2020	Gutierrez et al.	N/A	N/A
2022/0050061	12/2021	Bar	N/A	G03F 7/7065
2022/0067316	12/2021	Ackley	N/A	N/A
2022/0222500	12/2021	Ackley	N/A	N/A
2023/0041346	12/2022	Ackley et al.	N/A	N/A
2023/0125920	12/2022	D'Armancourt et al.	N/A	N/A
2023/0205466	12/2022	Gutierrez et al.	N/A	N/A
2023/0267287	12/2022	Ackley et al.	N/A	N/A
2024/0370677	12/2023	Ackley et al.	N/A	N/A

## FOREIGN PATENT DOCUMENTS

Patent No.	Application Date	Country	CPC
1085452	12/2000	EP	N/A
2270746	12/2010	EP	N/A
3336770	12/2017	EP	N/A
09-027049	12/1996	JP	N/A
2002-281287	12/2001	JP	N/A
2004-341764	12/2003	JP	N/A
4644283	12/2010	JP	N/A
2011-110777	12/2010	JP	N/A
2013-151126	12/2012	JP	N/A
10-0767433	12/2006	KR	N/A
2013/163789	12/2012	WO	N/A
2013/173985	12/2012	WO	N/A
2014/019130	12/2013	WO	N/A
2014/110495	12/2013	WO	N/A

## OTHER PUBLICATIONS

US 8,548,242 B1, 10/2013, Longacre (withdrawn) cited by applicant

US 8,616,454 B2, 12/2013, Havens et al. (withdrawn) cited by applicant

U.S. Appl. No. 18/146,201, filed Dec. 23, 2022, now U.S. Pat. No. 11,943,406, Patented. cited by applicant

U.S. Appl. No. 17/301,259, filed Mar. 30, 2021, now U.S. Pat. No. 11,570,321, Patented. cited by applicant

U.S. Appl. No. 16/790,417, filed Feb. 13, 2020, now U.S. Pat. No. 10,999,460, Patented. cited by applicant

U.S. Appl. No. 16/240,140, filed Jan. 4, 2019, now U.S. Pat. No. 10,834,283, Patented. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 21, 2020 for U.S. Appl. No. 16/240,295. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 23, 2021 for U.S. Appl. No. 16/930,022. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 26, 2022 for U.S. Appl. No. 17/301,259. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Dec. 21, 2022 for U.S. Appl. No. 17/301,259. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Dec. 31, 2020 for U.S. Appl. No. 16/790,417. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Feb. 8, 2023 for U.S. Appl. No. 17/468,425. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Feb. 9, 2023 for U.S. Appl. No. 17/454,148. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Feb. 27, 2023 for U.S. Appl. No. 17/468,425. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jan. 25, 2023 for U.S. Appl. No. 17/454,148. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jan. 27, 2021 for U.S. Appl. No. 16/790,417. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jan. 31, 2023 for U.S. Appl. No. 17/468,425. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jul. 14, 2022 for U.S. Appl. No. 17/301,259. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jul. 21, 2020 for U.S. Appl. No. 16/240,295. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jul. 28, 2020 for U.S. Appl. No. 16/240,140. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 1, 2023 for U.S. Appl. No. 17/454,148, 2 page(s). cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 2, 2020 for U.S. Appl. No. 16/240,295. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 8, 2021 for U.S. Appl. No. 16/930,022. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 8, 2022 for U.S. Appl. No. 17/301,259,. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 12, 2019 for U.S. Appl. No. 16/240,109. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 14, 2023 for U.S. Appl. No. 17/651,977, 8 page(s). cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Jun. 24, 2020 for U.S. Appl. No. 16/240,067. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Mar. 13, 2023 for U.S. Appl. No. 17/468,425. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Mar. 15, 2023 for U.S. Appl. No. 17/454,148. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Nov. 5, 2021 for U.S. Appl. No. 17/010,515. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Nov. 6, 2019 for U.S. Appl. No. 16/240,140. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Nov. 15, 2023 for U.S. Appl. No. 18/173,463, 9 page(s). cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Nov. 18, 2022 for U.S. Appl. No. 17/301,259. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Nov. 22, 2021 for U.S. Appl. No. 16/688,197. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Nov. 23, 2022 for U.S. Appl. No. 17/468,425. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Oct. 15, 2021 for U.S. Appl. No. 17/010,515. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Oct. 16, 2023 for U.S. Appl. No. 18/146,201, 9 page(s). cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 2, 2020 for U.S. Appl. No. 16/240,067. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 9, 2021 for U.S. Appl. No. 16/930,022. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 10, 2020 for U.S. Appl. No. 16/240,295. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 16, 2020 for U.S. Appl. No. 16/240,140. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 22, 2021 for U.S. Appl. No. 16/930,022. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 23, 2020 for U.S. Appl. No. 16/240,140. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 23, 2022 for U.S. Appl. No. 17/301,259. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 29, 2022 for U.S. Appl. No. 17/301,259. cited by applicant

Notice of Allowance and Fees Due (PTOL-85) Mailed on Sep. 29, 2023 for U.S. Appl. No. 17/651,977, 8 page(s). cited by applicant

Office Action for U.S. Appl. No. 14/824,455 dated Nov. 30, 2018, 17 pages. cited by applicant

Supplemental Notice of Allowability Mailed on Sep. 10, 2020 for U.S. Appl. No. 16/240,295. cited by applicant

U.S. Appl. No. 13/367,978 for a Laser Scanning Module Employing an Elastomeric U-Hinge Based Laser Scanning Assembly, filed Feb. 7, 2012, Feng et al. cited by applicant

U.S. Appl. No. 14/277,337 for Multipurpose Optical Reader, filed May 14, 2014, Jovanovski et al. cited by applicant

U.S. Appl. No. 14/283,282 for Terminal Having Illumination and Focus Control filed May 21, 2014, Liu et al. cited by applicant

U.S. Appl. No. 14/446,391 for Multifunction Point of Sale Apparatus With Optical Signature Capture filed Jul. 30, 2014, Good et al. cited by applicant

U.S. Appl. No. 14/676,109 for Indicia Reader, filed Apr. 1, 2015, Richard Walter Huck. cited by applicant

“Detecting Barcodes in Images with Python and OpenCV”[online] [retrieved on May 27, 2019] Retrieved from the Internet: <<https://www.pyimagesearch.com/2014/11/24/detecting-barcodes-images-python-opencv/>> dated Nov. 24, 2014. cited by applicant

Anonymous: “How can I quantify difference between two images?—Stack Overflow”, [online] [retrieved on May 27, 2019] Retrieved from the Internet: URL:<<https://stackoverflow.com/questions/189943/how-can-i-quantify-difference-between-two-images>> dated Apr. 21, 2015, pp. 1-5. cited by applicant

Basic Image Enhancement and Analysis Techniques, 4 pages, [online], [retrieved on Oct. 3, 2016]. Retrieved from the Internet <URL:<<https://in.mathworks.com/help/images/image-enhancement-and-analysis.html>>. cited by applicant

Communication about intention to grant a European patent Mailed on Dec. 23, 2022 for EP Application No. 19150493.5. cited by applicant

Communication about intention to grant a European patent received for EP Application No. 19150495.0, mailed on Nov. 15, 2021, 6 pages. cited by applicant

Communication Pursuant to Article 94(3) issued in European Application No. 19150493.5 on Jul. 15, 2021, 7 pages. cited by applicant

Communication Pursuant to Article 94(3) issued in European Application No. 19150495.0 on Jul. 9, 2021, 6 pages. cited by applicant

Communication pursuant to Rules 70(2) and 70a(2) for European Application No. 19150494.3, dated Jul. 15, 2019, 2 pages. cited by applicant

Communication pursuant to Rules 70(2) and 70a(2) for European Application No. 19150495.0, dated Jul. 15, 2019, 2 pages. cited by applicant

Decision to grant a European patent Mailed on May 8, 2023 for EP Application No. 19150493, 2 page(s). cited by applicant

Decision to grant a European patent Mailed on Sep. 14, 2023 for EP Application No. 19150494, 2 page(s). cited by applicant

Decision to grant a European patent received for European Application No. 19150495.0, mailed on

Mar. 24, 2022, 2 pages. cited by applicant  
EP Office Action Mailed on Aug. 13, 2021 for EP Application No. 19150494. cited by applicant  
EP Office Action Mailed on Feb. 20, 2023 for EP Application No. 19150494. cited by applicant  
European search opinion Mailed on Jun. 4, 2019 for EP Application No. 19150495.0, 5 pages. cited by applicant  
European search opinion Mailed on Jun. 6, 2019 for EP Application No. 19150494. cited by applicant  
European search report and Search opinion Mailed on May 24, 2019 for EP Application No. 19150493.5. cited by applicant  
European search report Mailed on Jun. 4, 2019 for EP Application No. 19150495, 2 page(s). cited by applicant  
European search report Mailed on Jun. 6, 2019 for EP Application No. 19150494. cited by applicant  
Examiner initiated interview summary (PTOL-413B) Mailed on Apr. 16, 2020 for U.S. Appl. No. 16/240,067. cited by applicant  
Extended European Search Report Mailed on Dec. 21, 2023 for EP Application No. 23195821, 8 page(s). cited by applicant  
Extended European search report Mailed on Jun. 22, 2022 for EP Application No. 22160775.7, 7 pages. cited by applicant  
Extended European Search Report Mailed on Jun. 29, 2023 for EP Application No. 23163411, 7 page(s). cited by applicant  
Faulty Barcode Detection, 11 pages, [online], [retrieved on Oct. 24, 2016]. Retrieved from the Internet <URL:<http://tewson.com/sites/default/files/barcode.pdf>>. cited by applicant  
Final Rejection Mailed on Jan. 10, 2020 for U.S. Appl. No. 16/240,067. cited by applicant  
Find Image Rotation and Scale Using Automated Feature Matching, 7 pages, [online], [retrieved on Oct. 3, 2016]. Retrieved from the Internet <URL: [https://www.mathworks.com/examples/matlab-computer-vision/mw/vision\\_product-visionrecovertform-find-image-rotation-and-scale-using-automated-feature-matching](https://www.mathworks.com/examples/matlab-computer-vision/mw/vision_product-visionrecovertform-find-image-rotation-and-scale-using-automated-feature-matching)>. cited by applicant  
Image Analysis, 1 page, [online], [retrieved on Nov. 7, 2016]. Retrieved from the Internet <URL: <http://in.mathworks.com/help/images/image-analysis.html>>. cited by applicant  
Image Enhancement, 1 page, [online], [retrieved on Oct. 3, 2016]. Retrieved from the Internet <URL: <http://in.mathworks.com/help/images/image-enhancement-and-restoration.html>>. cited by applicant  
Imshowpair—compare differences between images, 6 pages, [online], [retrieved on Nov. 7, 2016]. Retrieved from the Internet <URL: <http://in.mathworks.com/help/images/ref/imshowpair.html?requestedDomain=in.mathworks.com#bta3zrg>>. cited by applicant  
Intention to grant Mailed on Jun. 5, 2023 for EP Application No. 19150494, 9 page(s). cited by applicant  
Non-Final Rejection Mailed on Aug. 2, 2022 for U.S. Appl. No. 17/468,425, 8 pages. cited by applicant  
Non-Final Rejection Mailed on Aug. 25, 2020 for U.S. Appl. No. 16/790,417. cited by applicant  
Non-Final Rejection Mailed on Dec. 21, 2023 for U.S. Appl. No. 18/306,904, 7 page(s). cited by applicant  
Non-Final Rejection Mailed on Feb. 2, 2023 for U.S. Appl. No. 17/651,977. cited by applicant  
Non-Final Rejection Mailed on Feb. 8, 2022 for U.S. Appl. No. 17/301,259. cited by applicant  
Non-Final Rejection Mailed on Jul. 11, 2019 for U.S. Appl. No. 16/240,067. cited by applicant  
Non-Final Rejection Mailed on Jul. 27, 2023 for U.S. Appl. No. 18/173,463, 8 page(s). cited by applicant  
Non-Final Rejection Mailed on Jun. 26, 2019 for U.S. Appl. No. 16/240,140. cited by applicant  
Non-Final Rejection Mailed on Mar. 12, 2020 for U.S. Appl. No. 16/240,140. cited by applicant

Non-Final Rejection Mailed on May 2, 2019 for U.S. Appl. No. 16/240,109. cited by applicant  
Non-Final Rejection Mailed on May 11, 2023 for U.S. Appl. No. 18/146,201, 14 page(s). cited by applicant  
Non-Final Rejection Mailed on May 26, 2021 for U.S. Appl. No. 16/688,197. cited by applicant  
Non-Final Rejection Mailed on Nov. 27, 2020 for U.S. Appl. No. 16/688,197. cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Apr. 12, 2024 for U.S. Appl. No. 18/306,904, 8 page(s). cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Apr. 16, 2020 for U.S. Appl. No. 16/240,067. cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 3, 2023 for U.S. Appl. No. 17/454,148, 2 page(s). cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 7, 2020 for U.S. Appl. No. 16/240,140. cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 11, 2021 for U.S. Appl. No. 17/010,515. cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 12, 2022 for U.S. Appl. No. 17/301,259. cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Aug. 14, 2019 for U.S. Appl. No. 16/240,109. cited by applicant  
Notice of Allowance and Fees Due (PTOL-85) Mailed on Mar. 12, 2025 for U.S. Appl. No. 18/771,938, 8 page(s). cited by applicant  
EP Office Action Mailed on Apr. 16, 2025 for EP Application No. 23195821, 7 page(s). cited by applicant  
Communication about intention to grant a European patent Mailed on May 16, 2025 for EP Application No. 22160775, 6 page(s). cited by applicant  
Communication about intention to grant a European patent Mailed on Jun. 30, 2025 for EP Application No. 23163411, 6 page(s). cited by applicant

---

*Primary Examiner:* Popovici; Dov

*Attorney, Agent or Firm:* Alston & Bird LLP

---

## **Background/Summary**

CROSS-REFERENCE TO RELATED APPLICATIONS (1) This application is a continuation of U.S. patent application Ser. No. 18/146,201, filed Dec. 23, 2022, which is a continuation of U.S. patent application Ser. No. 17/301,259, filed Mar. 30, 2021 (now U.S. Pat. No. 11,570,321 issued Jan. 31, 2023), which is a continuation of U.S. patent application Ser. No. 16/790,417, filed Feb. 13, 2020 (now U.S. Pat. No. 10,999,460 issued May 4, 2021), which is a continuation of U.S. patent application Ser. No. 16/240,140, filed Jan. 4, 2019 (now U.S. Pat. No. 10,834,283 issued Nov. 10, 2020), which claims the benefit of U.S. Provisional Patent Application No. 62/614,089, filed Jan. 5, 2018, the entire contents of each are incorporated herein by reference.

## **FIELD OF THE INVENTION**

(1) The present invention relates to printing, and more particularly to visual printing defect detection and analysis.

## **BACKGROUND**

(2) Generally speaking, both thermal direct and thermal transfer printers are the preferred technology for printing barcodes. However, printouts may exhibit visible printing defects that may

impact the ANSI grading or visual output of the labels. For example, the produced output is often plagued by printing defects, such as “ribbon wrinkle,” “white banding,” “black banding,” “voids,” or “ink split.” Such problems may stem from media and printhead contamination, improper ribbon setup, printer wear and tear, uneven pressure, mechanical design margin, and other triggers that may not be related to thermal management algorithms. Printing defect degrades label print quality, and may lead to unusable media. Such issues may lead to equipment damage and supply waste.

#### SUMMARY

(3) Accordingly, the present invention embraces methods for visual printing defect detection.

(4) In accordance with various embodiments of the preset invention, a method for printing defect detection is provided. The method comprises scanning an image with a verifier to generate a scanned image; processing the scanned image to obtain a difference image by comparing the scanned image to a reference image and match parameters of the scanned image to the reference image; analyzing the difference image to detect one or more printing defects; grouping the one or more detected printing defects in one or more groups; refining the grouping of the detected printing defects; assigning confidence level values to the one or more refined groups; and performing a confidence level analysis to determine if one or more servicing actions should be performed.

(5) In some embodiments, refining the grouping includes performing image processing of the difference image to detect at least one bright pixel common for two or more groups.

(6) In some embodiments, the method further comprises extracting and/or reassigning one or more pixels to a correct group.

(7) In some embodiments, the confidence level is calculated based on a ratio of a total amount of bright pixels in a selected area over a total amount of pixels in that area.

(8) In some embodiments, performing a confidence level analysis includes performing a confidence level history analysis.

(9) In some embodiments, the method further comprises analyzing the scanned image to determine presence or absence of printhead, platen roller and/or media contamination.

(10) In some embodiments, the method further comprises determining a type of a printing defect by comparing the assigned confidence level to a predetermined table of confidence levels.

(11) In some embodiments, performing a confidence level analysis includes calculating confidence level values for one or more defect groups, and monitoring changes in the calculated values from one or more previous images to the current scanned image.

(12) In some embodiments, scanning an image with a verifier includes scanning an image displaying a barcode symbol, text, and/or graphics.

(13) In accordance with various embodiments of the preset invention, a method for printing defect analysis is provided. The method comprises capturing an image of a printout on a media; checking the captured image for one or more printing defects; analyzing evolution of the detected printing defects between the current captured image and one or more images of preceding printouts; and using results of the evolution analysis to determine if one or more predetermined corrective actions should be initiated.

(14) In some embodiments, checking for printing defects includes generating and analyzing a difference image obtained by comparing the captured image to a reference image, and/or processing the captured image to detect printhead, platen roller and/or media contamination.

(15) In some embodiments, generating a difference image includes comparing a binary version of the captured image to a binary version of the reference image.

(16) In some embodiments, checking the captured image for one or more printing defects includes detecting at least one of ribbon wrinkles, printhead, platen roller and/or media contamination, black and/or white banding, and/or black and/or white ink split.

(17) In some embodiments, analyzing evolution includes comparing one or more evolution characteristics of the defects to a predetermined threshold.

(18) In some embodiments, the method further comprises verifying the one or more evolution



characteristics when one or more characteristics are found to exceed the predetermined threshold, and determining if one or more predetermined corrective actions should be initiated.

(19) In some embodiments, the method further comprises initiating one or more predetermined corrective actions when the one or more evolution characteristics are slowly rising without exceeding the predetermined threshold. In some embodiments, initiating corrective actions includes triggering an alert, producing an error message, stopping printer operation, and/or prescribing a recommended course of action.

(20) In some embodiments, the method further comprises providing feedback to a self-learning defect database. In some embodiments, the method further comprises using the provided feedback to dynamically update one or more algorithms for checking the image for printing defects, and/or for analyzing evolution of the detected defects.

(21) In some embodiments, capturing an image of a printout on a media includes scanning an image with a printer-verifier device.

(22) In accordance with various embodiments of the preset invention, a method for generating a difference image is provided. The method comprises adjusting position, size, and luminance of a scanned image with respect to a reference image, such as aligning a scanned image with a reference image, scaling the scanned image to match a size of the reference image, and adjusting luminance of the scaled scanned image; conducting a pixel-to-pixel analysis between the scanned image and the reference image; and generating a difference image based on results of the pixel-to-pixel analysis.

(23) In some embodiments, the method further comprises obtaining the scanned image with a printer-verifier device.

(24) In some embodiments, the scanned image includes stretching or compressing the scanned image in a transverse direction and/or longitudinal direction. In some embodiments, the method further comprises analyzing the difference image to identify one or more printing errors.

(25) In some embodiments, identifying printing errors includes detecting at least one of ribbon wrinkles, printhead and/or media contamination, platen roller contamination, black and/or white banding, and/or black and/or white ink split.

(26) In accordance with various embodiments of the preset invention, a method for image processing is provided. The method comprises normalizing a size of a produced image to match a size of a reference image; normalizing a luminance of the produced image to match a luminance of the reference image; and producing a difference image by comparing the normalized produced image to the reference image.

(27) In some embodiments, normalizing a luminance includes performing level adjustment to match white and/or black colors in the produced image and white and/or black colors in the reference image. In some embodiments, normalizing a luminance includes performing level adjustment to match red, green and/or blue colors in the produced image and red, green and/or blue colors in the reference image. In some embodiments, normalizing a size of a produced image includes adjusting a size of the produced image to align corners and/or edges of the produced image with corners and/or edges of the reference image. In some embodiments, normalizing a size of a produced image includes equalizing a distance between a rightmost printed area and a leftmost printed area of the image.

(28) In some embodiments, comparing the normalized produced image to the reference image includes comparing information of the produced image to a print command string.

(29) In some embodiments, the method further includes detecting one or more printing defects.

(30) In some embodiments, detecting one or more printing defects includes detecting at least one of ribbon wrinkles, printhead and/or media contamination, black and/or white banding, and/or black and/or white ink split.

(31) In some embodiments, the method further includes capturing the produced image with an integrated verifier device.

- (32) In accordance with various embodiments of the preset invention, a method for print defect detection is provided. The method comprises scanning an image with a verifier to generate a captured image; processing the captured image to match one or more parameters of the captured image to one or more parameters of a reference image; comparing the captured image to the reference image to detect one or more bright pixels; and detecting one or more print defects.
- (33) In some embodiments, processing the captured image includes scaling, rotating, adjusting luminance and/or adjusting one or more colors.
- (34) In some embodiments, comparing the captured image to the reference image includes comparing a binary version of the captured image to a binary version of the reference image.
- (35) In some embodiments, the method further comprises performing a bright pixel analysis. In some embodiments, performing a bright pixel analysis includes grouping and/or connecting bright pixels located within a predetermined distance from each other.
- (36) In some embodiments, capturing an image with a verifier includes capturing an image displaying a barcode symbol, text, and/or graphics.
- (37) In accordance with various embodiments of the preset invention, a method for determining an origin of printing distortion is provided. The method includes generating an image of a printed barcode symbol with a printer-verifier; processing the image to detect deviation of parameters of elements of the barcode symbol from a predetermined threshold; checking for a malfunction of heating elements; and determining an origin of printing distortion.
- (38) In some embodiments, wherein processing the image of the printed barcode symbol includes calculating widths of the one or more elements of the barcode symbol. In some embodiments, the method further comprises averaging the calculated widths for a group of the elements of the barcode symbol. In some embodiments, the method further comprises plotting the averaged widths.
- (39) In some embodiments, processing the image of the printed barcode symbol includes comparing the image with a reference image.
- (40) In some embodiments, checking for a malfunction of one or more heating elements includes checking for a burnout of one or more heating elements.
- (41) In some embodiments, processing the image of the printed barcode symbol includes processing the image of the printed barcode symbol with an image processor.
- (42) In accordance with various embodiments of the preset invention, the present invention embraces methods for automated detection of a contaminated printhead.
- (43) In accordance with various embodiments of the preset invention, a method for detecting a contaminated thermal printhead is provided. The method includes identifying a barcode symbol in a verifier image, and analyzing its scan lines; calculating element widths of the scan lines using a predetermined threshold; and analyzing the calculated element widths to detect contamination of a thermal printhead.
- (44) In some embodiments, identifying a barcode symbol in a verifier image includes identifying a barcode symbol in an image obtained from an integrated printer-verifier device.
- (45) In some embodiments, analyzing the calculated element widths includes plotting an average deviation within a group of the element widths.
- (46) In some embodiments, calculating element widths includes calculating widths of one or more narrow bars.
- (47) In some embodiments, calculating element widths includes calculating widths of one or more narrow spaces between bars of the barcode.
- (48) In some embodiments, the method further comprises issuing a notification reporting the detected contamination, and prescribing a recommended course of action.
- (49) In accordance with various embodiments of the preset invention, a method for detecting printhead contamination is provided. The method includes detecting one or more indicia in an image of a printed image; analyzing elements of the detected indicia; conducting a pattern match analysis between the detected indicia and a reference image to produce a difference image; and

analyzing the difference image to detect a presence of printhead contamination.

(50) In some embodiments, detecting one or more indicia in an image of a printed image includes detecting indicia using a verifier integrated into a thermal printer. In some embodiments, detecting one or more indicia includes detecting one or more 1D barcodes, 2D barcodes, graphics, and/or text.

(51) In some embodiments, the method including reporting the presence of printhead contamination. In some embodiments, reporting the presence of printhead contamination includes sending a message to an entity responsible for printhead maintenance. In some embodiments, reporting the presence of printhead contamination includes printing a specifically formatted label.

(52) In some embodiments, the method further comprises triggering one or more predetermined actions in response to the detected contamination, wherein the predetermined actions are selected based on user sensitivity to print quality.

(53) In accordance with various embodiments of the preset invention, an image processing method is provided. The method includes capturing an image of a label; processing the captured image to produce a difference image having a plurality of bright pixels by comparing the captured image to a reference image; consecutively connecting the bright pixels located within a predetermined radius to form a line until there are no pixels within the radius left to connect; and iteratively connecting the pixels until all the pixels of the plurality of bright pixels having neighboring pixels within the predetermined radius are connected, and one or more lines are formed.

(54) In some embodiments, consecutively connecting the pixels further includes monitoring a running average slope defining an orientation of the line being formed.

(55) In some embodiments, monitoring a running average slope includes determining an angle of the slope with a point-slope technique.

(56) In some embodiments, the method further comprises making a connection between two consecutive pixels when a resulting change in the running average slope does not exceed a predetermined angle threshold.

(57) In some embodiments, the method further comprises making a connection between two consecutive pixels when a resulting change in the running average slope does not exceed a predetermined dynamic angle value.

(58) In some embodiments, the method further comprises monitoring an average direction of the running average slope, and making a connection between two consecutive pixels when such connection follows a forward direction of the slope.

(59) In accordance with various embodiments of the preset invention, a method for determining a ribbon wrinkle is provided. The method includes creating a difference image to locate one or more bright points by comparing a captured image of a media after printing to a reference image; grouping the bright points located near each other to form one or more primary lines characterized by a running average slope; and connecting the primary lines having a similar running average slope to form one or more secondary lines.

(60) In some embodiments, the method further comprises assigning a confidence level value to the one or more primary and/or secondary lines. In some embodiments, the method further comprises requesting capturing an additional image to replace the captured image having one or more primary and/or secondary lines with low confidence level values.

(61) In some embodiments, creating a difference image includes using the reference image stored in a self-learning database.

(62) In some embodiments, using the reference image stored in a self-learning database includes using the reference image stored in an external database.

(63) In some embodiments, comparing a captured image of a media includes comparing a captured image of a media displaying a barcode.

(64) In accordance with various embodiments of the preset invention, a ribbon wrinkle detection method is provided. The method includes identifying a barcode symbol having a plurality of

elements displayed on a media; surrounding the barcode symbol with a bounding box encompassing top and bottom parts and outer edges of the barcode symbol, and/or one or more finder patterns; locating one or more unprinted points located near the elements of the barcode symbol; connecting co-localized unprinted points to form one or more lines; determining an angle of the one or more lines relative to the bounding box; and verifying that each of the determined angles exceeds a predetermined threshold value.

(65) In some embodiments, identifying a barcode symbol includes identifying a two-dimensional barcode symbol.

(66) In some embodiments, the method further comprises determining a number of the one or more lines. In some embodiments, the method further comprises displaying a result of the angle verification.

(67) In some embodiments, determining an angle includes determining an angle with a point-slope technique. In some embodiments, determining an angle includes determining an angle with a linear regression technique. In some embodiments, verifying that each of the determined angles exceeds a predetermined threshold value includes exceeding a predetermined dynamic angle value. In some embodiments, verifying that each of the determined angles exceeds a predetermined threshold value includes exceeding a predetermined fixed angle value.

(68) In accordance with various embodiments of the preset invention, the present invention embraces methods of detecting platen roller contamination. In an embodiment, the method for determining a platen roller contamination can include creating a difference image to locate one or more bright points by comparing a captured image of a media after printing to a reference image, and grouping the bright points located near each other to form one or more voids. Additionally, the method can include analyzing the void pattern, and/or determining whether the contamination is on the platen roller or on the media and/or ribbon.

(69) In accordance with various embodiments of the preset invention, a platen roller contamination detection method is provided. The method comprises capturing an image of a label; processing the captured image to produce a difference image having a plurality of bright pixels by comparing the captured image to a reference image; consecutively connecting the bright pixels located within a predetermined radius to form a cluster and detecting a void mark, until there are no pixels within the radius left to connect; and iteratively connecting the pixels until all the pixels of the plurality of bright pixels having neighboring pixels within the predetermined radius are connected, and one or more void marks are detected.

(70) In some embodiments, consecutively connecting the bright pixels includes grouping each cluster of pixels depending on proximity between neighboring bright pixels, a slope change between clusters of pixels, and a void space existing between the clusters of pixels.

(71) In some embodiments, the method further comprises determining a repeat pattern of the detected one or more void marks at a defined distance along a vertical line, wherein the defined distance corresponds to circumference of the platen roller.

(72) In some embodiments, the method further comprises issuing a priority message of a first level for an operator in an instance when a count of the one or more void marks are less than a threshold value. In some embodiments, the method further comprises initiating a blast of compressed air in a defined proximity to the platen roller in an instance when the count of the one or more void marks exceed the threshold value.

(73) In some embodiments, detecting the void mark includes identifying one or more areas of the bright pixels falling within a defined circular area, wherein in accordance with a first criteria, separation of two bright pixels is more than a defined threshold separation in an instance when a count of the bright pixels is one or more, wherein in accordance with a second criteria, no three bright pixels can lie on a straight line in an instance when the count of the bright pixels is within a first range, wherein in accordance with a third criteria, no group of five bright pixels can lie on the straight line when the count of the bright pixels is beyond the first range, wherein, in an instance

when the first, second and third criteria are met, a printed label is deemed to have defects caused by a contaminated platen roller.

(74) The foregoing illustrative summary, as well as other exemplary objectives and/or advantages of the invention, and the manner in which the same are accomplished, are further explained within the following detailed description and its accompanying drawings.

---

## Description

### BRIEF DESCRIPTION OF THE DRAWINGS

(1) The description of the illustrative embodiments may be read in conjunction with the accompanying figures. It will be appreciated that for simplicity and clarity of illustration, elements illustrated in the figures have not necessarily been drawn to scale. For example, the dimensions of some of the elements are exaggerated relative to other elements. Embodiments incorporating teachings of the present disclosure are shown and described with respect to the figures presented herein, in which:

- (2) FIG. 1 schematically depicts a defect detection algorithm, according to an embodiment;
- (3) FIG. 2A graphically depicts a diagram of rescaling of a captured image, according to an embodiment;
- (4) FIG. 2B graphically depicts a representative luminance range of reference and captured images, according to an embodiment;
- (5) FIG. 3 schematically depicts a method for generating a difference image, according to an embodiment;
- (6) FIG. 4 schematically depicts a method for image processing, according to an embodiment;
- (7) FIG. 5 schematically depicts a method for print defect detection, according to an embodiment;
- (8) FIG. 6A schematically depicts an exemplary embodiment of a portion of a typical barcode printed with printhead contamination;
- (9) FIG. 6B graphically depicts a plot of average narrow bars widths across the barcode shown in FIG. 6A;
- (10) FIG. 6C schematically depicts an exemplary embodiment of pattern match analysis for the barcode shown in FIG. 6A;
- (11) FIG. 6D schematically depicts an exemplary embodiment of a printed label produced with a speck of dust physically adhered to a printhead;
- (12) FIG. 7 schematically depicts a method for determining an origin of printing distortion, according to an embodiment;
- (13) FIG. 8 schematically depicts a method for detecting a contaminated thermal printhead, according to an embodiment;
- (14) FIG. 9 schematically depicts a method for detecting printhead contamination, according to an embodiment;
- (15) FIG. 10 schematically depicts an image processing method, according to an embodiment;
- (16) FIG. 11 schematically depicts a method for determining a ribbon wrinkle, according to an embodiment;
- (17) FIG. 12A schematically depicts a ribbon wrinkle detection method, according to an embodiment;
- (18) FIG. 12B graphically depicts a 2D barcode symbol (left) and the 2D barcode symbol surrounded with a bounding box (right);
- (19) FIG. 12C graphically depicts a linear barcode symbol (top) and the linear barcode symbol surrounded with a bounding box (bottom);
- (20) FIG. 13A graphically depicts a printed label having distortions caused by a ribbon wrinkle;
- (21) FIG. 13B graphically depicts a difference image of the label of FIG. 13A, according to an

embodiment;

(22) FIG. 13C graphically depicts a pattern made by connecting dots in the difference image of FIG. 13B, according to an embodiment;

(23) FIG. 13D graphically depicts a portion of a difference image with three separately detected wrinkle lines having a similar running average slope, according to an embodiment;

(24) FIG. 13E graphically depicts a line formed by connecting the three lines of FIG. 13D, according to an embodiment;

(25) FIG. 14A graphically depicts a printed label having distortions caused by a contaminated platen roller;

(26) FIG. 14B graphically depicts a difference image of the label of FIG. 14A, according to an embodiment;

(27) FIG. 14C graphically depicts a relationship between bright points in the difference image of FIG. 14B, according to an embodiment;

(28) FIG. 14D graphically depicts a relationship between four different voids (left) and a zoomed-in image of one of the voids (right), according to an embodiment;

(29) FIG. 14E graphically depicts repeated void marks on a label printed with a contaminated platen roller;

(30) FIG. 15 depicts an example of a label printed with a combination of a void and banding defects;

(31) FIG. 16 depicts an example of a label printed with wrinkle and banding defects;

(32) FIGS. 17A-17D depict an example of a label containing wrinkle and banding defects before and after refinement;

(33) FIG. 18 depicts a group information data used for confidence level value calculation, according to an embodiment;

(34) FIG. 19 schematically depicts a history algorithm, according to an embodiment;

(35) FIG. 20 schematically depicts a method for printing defect analysis, according to an embodiment;

(36) FIG. 21A graphically illustrates a portion of an exemplary printer-verifier (a cover of the printer-verifier is removed to illustrate an interior thereof), according to an embodiment;

(37) FIG. 21B schematically depicts a block diagram of the printer-verifier of FIG. 21A, according to an embodiment; and

(38) FIG. 22 schematically depicts an exemplary printer communicatively coupled to a verifier in a system for detecting printing defects, according to an embodiment.

## DETAILED DESCRIPTION OF THE INVENTION

(39) In accordance with various embodiments of the present invention, methods, apparatuses, and systems for visual printing defect detection and analysis are provided.

(40) Some embodiments of the present disclosure will now be described more fully hereinafter with reference to the accompanying drawings, in which some, but not all embodiments of the disclosure are shown. Indeed, these disclosures may be embodied in many different forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided so that this disclosure will satisfy applicable legal requirements. Like numbers refer to like elements throughout.

(41) Unless the context requires otherwise, throughout the specification and claims which follow, the word “comprise” and variations thereof, such as, “comprises” and “comprising” are to be construed in an open sense, that is as “including, but not limited to.”

(42) Reference throughout this specification to “one embodiment” or “an embodiment” means that a particular feature, structure or characteristic described in connection with the embodiment is included in at least one embodiment. Thus, the appearances of the phrases “in one embodiment” or “in an embodiment” in various places throughout this specification are not necessarily all referring to the same embodiment. Furthermore, the particular features, structures, or characteristics may be

combined in any suitable manner in one or more embodiments.

(43) The word “example” or “exemplary” is used herein to mean “serving as an example, instance, or illustration.” Any implementation described herein as “exemplary” is not necessarily to be construed as preferred or advantageous over other implementations.

(44) If the specification states a component or feature “may,” “can,” “could,” “should,” “would,” “preferably,” “possibly,” “typically,” “optionally,” “for example,” “often,” or “might” (or other such language) be included or have a characteristic, that a specific component or feature is not required to be included or to have the characteristic. Such component or feature may be optionally included in some embodiments, or it may be excluded.

(45) The headings provided herein are for convenience only and do not limit the scope or meaning of the claimed invention.

(46) Various embodiments of the present invention will be described in relation to a thermal transfer printer. However, the present invention may be equally applicable to other types and styles of printers (inclusive of printer-verifiers) (e.g., a thermal direct printer, a laser toner printer, an ink drop printer, etc.).

(47) As used herein, the terms “media” and “print media” include label stock, label, ribbon, card stock, and other materials used in a printing process, such as a thermal transfer printing process or a direct thermal printing process.

(48) As used herein, the terms “target image,” “reference image,” and “buffer image” may be used interchangeably and considered synonymous depending on the context, unless further definition is provided.

(49) As used herein, the terms “scanned image,” “produced image,” “blend image,” “verifier image,” and “captured image” may be used interchangeably and considered synonymous depending on the context, unless further definition is provided.

(50) As used herein, the terms “pixel,” “dot,” and “point” may be used interchangeably and considered synonymous depending on the context, unless further definition is provided.

## I. Overview

(51) Generally speaking, thermal printing has become the dominant means of generating barcode labels on demand. Thermal printers can include any printing technology that utilizes a “print command string,” such as thermal direct, laser toner, ink drop, sublimation, impact dot matrix, and thermochromic and, by reverse image, phototypeset, screen, gravure and offset. Over time, printheads get contaminated, causing reduced print quality and increasing wear, which in turn can shorten printhead life. Preventative maintenance may be instituted; however, this can be wasteful when the printhead does not need cleaning, or worse, can be damaging when the printhead needs cleaning prior to the scheduled maintenance. Further, in many situations, proper cleaning is ignored until printer failure. Therefore, a need exists for an automated method of detecting a contaminated thermal printhead so that the issue could be resolved in a timely fashion.

(52) Further, platen roller contamination can significantly affect print quality of labels, rendering them defective and unusable. For example, printing labels in industrial settings can lead to dust, grit and other airborne particulate getting into the printer components. Making matters worse, this contamination is often statically charged, causing it to be especially attracted to polymers, which form the basis of the media-moving assemblies (e.g., the platen rollers) in the printers. Therefore, a need exists for an automated method for detecting platen roller contamination to overcome this issue.

(53) Moreover, media contamination can also lead to the printing quality reduction. Ribbon wrinkle may be caused by a setup of the printer and/or loading of media and/or ribbon. Additionally, the amount of heat applied or the appearance of the label pattern may also affect the printing quality. For example, printing only on the side of the label may cause the ribbon to pull unevenly and wrinkle. When the heat from the printhead does not adequately get to the media due to a wrinkle or contamination of printer parts (such as a platen roller) and/or media or ribbon contamination, a part

of the intended image is not created. This can look like an unprinted dot, void or a line in an otherwise printed area. When the resulting label print quality is degraded due to such contamination or a ribbon wrinkle, these problems need to be identified and addressed. Therefore, a need exists for an automated method for detecting and correction of ribbon wrinkle.

(54) In addition to printhead, platen roller, and media contamination described above, other issues may exist that may cause printing defects. To handle such issues more efficiently, the printing process should utilize an algorithm to automatically detect a wide variety of visual printing defects, followed by providing user notification, guidance and/or taking actions to resolve the detected issues.

(55) Although some attempts have been made to create printing defect detection, none of the proposed methods combines determining a quality grading of a label immediately after printing with performing printing diagnostics not covered by the grading criteria, or focuses on creating an exact match to the reference image, or covers a reference image represented as a print command string.

(56) Therefore, a need exists for a method capable of combining automatic detection of various visual printing defects (such as banding, void, ribbon wrinkle, ink split, and printhead, media and/or platen roller contamination) that may occur during the printout, alerting the user in real-time, and providing advice on the corrective actions.

## II. Printing Defect Detection

(57) FIG. 1 depicts an example implementation of a defect detection algorithm, in which a printer-scanned image (that may be obtained with a verifier) is compared to an original reference image (e.g., a graphically rearranged print command string) in order to highlight the printout difference between the two (as described further hereinafter).

(58) Referring to FIG. 100, at step 102, a printed image may be scanned. In an embodiment, an image, such as a printer-scanned image, may be obtained with a verifier. In an embodiment, scanning an image with a verifier can include scanning an image displaying a barcode symbol, text, and/or graphics. The image may be captured by any method of digital image capture, such as imagers, image sensors (such as an integrated 1D, 2D, or 3D image sensor), verifiers, fax scanners and cameras. Various examples of the image sensors may include, but are not limited to, a contact image sensor (CIS), a charge-coupled device (CCD), or a complementary metal-oxide semiconductor (CMOS) sensor. For example, the captured image can be created by an optical head that captures sequential digital “snap shots” of a single line traversing the label's width as the media moves through the printer. Such image may be slightly wider and/or longer than the reference image due to optical effects, such as aperture distortion. There may also be distortions caused by the imperfect movement of the media through the printer. Such an image should be processed with a scaling algorithm so that the captured and reference images are exactly of the same size. If there is a perfectly printed label, a binarized version of the captured image will be indistinguishable from a printout of the bitmap of the reference image.

(59) In an embodiment, at step 104, a contaminated thermal printhead may be automatically determined based on the printer-scanned image, as described in FIGS. 6A to 9.

(60) In accordance with another embodiment, at step 106, the captured image may be compared with the reference image, such as a print command string for determining a difference image to detect one or more printing defects. The print command string (which is computer code) can be shown graphically (where a “to print” command is depicted as black and a “no-print” command is depicted as white) and then rearranged into rows and columns to mimic approximately what the printed image will look like. The graphically rearranged print command string can be referred to as the “reference image.” After printing a label or tag, the printed image can be optically scanned where light is reflected off the printed image and captured, using one or more photosensitive semiconductor elements producing a captured image of the printed matter.

(61) The captured image may have a slight stretch/compression or a small tilt in the image



compared to the reference image. In this case, the first step can include “re-aligning” the captured image with the reference image. A scaling/rotation algorithm may be used to ensure that the corners and edges of the images align with each other if the two images were to overlap. Width and length of the captured image may differ from the reference image by different factors.

(62) The scaling algorithm that can be applied to the captured image in the horizontal (across the web) direction can be configured to compress the image so that the distance between the leftmost printed area and the rightmost printed area is equal. For example, the size of the reference image is known exactly because it maps directly into the dots of the printhead. A typical printhead may be described as having 300 dots per inch (dpi), whereas the actual size of the dots is 0.0833 mm (often referred to as 12 dots per mm). Optical scanner such as a verifier often differs from this number, and is typically 24 pixels per mm. Combined with optical distortions, this can yield a different value of the printing width. A horizontal scaling algorithm can be based on the following equations: Processed captured image=Reference image dimension/Original captured image dimension\*Every original captured image pixel;

Processed captured image=the above result digitized into 12 dots per mm.

(63) The result is a pixel-to-pixel comparison between the captured and reference images. Similar scaling algorithm can be applied for the vertical direction (in the direction of the web motion), except that the captured image may be longer or shorter due to mechanical effects in addition to the optical effects. The determination of the difference image is further described in FIGS. 2A to 5.

(64) In an embodiment, comparing the scanned image to the reference image may include comparing a binary version of the scanned image to a binary version of the reference image to detect bright pixels marking the areas where the two images do not match.

(65) Based on the determination of the difference image, grouping of the detected bright pixels (also referred to as difference pixels) may be performed. In other words, the one or more the detected printing defects may be grouped. In an embodiment, grouping of the detected difference pixels may be performed for printhead contamination at step **104** (further described in FIGS. 6A to 9). In another embodiment, grouping of the detected difference pixels may be performed for wrinkles and void detection at steps **108** and **110** (further described in FIGS. 10 to 14E). In another embodiment, grouping of the detected difference pixels may be performed for banding detection at step **112** (covering horizontal, vertical white and black banding) (further described in FIGS. 15 to 17D).

(66) At step **114**, the grouping data is then refined to ensure that bright pixels belong to correct defect types. The grouping data may be refined by performing image processing of the difference image to detect at least one bright pixel common for two or more groups. For example, a separate difference image can be created for each detected defect type (void, wrinkle, banding, ink split, etc.). Such difference images can be generated to have the same size (length and width), and consequently the same total number of pixels. Each difference image contains coordinates (unique x and y positions within the image) of each of the bright pixels detected. Thus, refining the grouping can include determining whether a particular bright pixel is common for any of the difference images obtained by comparing the scanned image to the reference image. For example, if a pixel in row 17, column 28 is defined as “bright” in the void difference image, but as “dark” in the wrinkle difference image, then it is not a common pixel. But if the pixel is defined as bright in both images, then it is a common pixel. The method **100** can further include extracting and/or reassigning one or more pixels to a correct group.

(67) Step **114** is especially beneficial for defect analysis when multiple defects occur simultaneously and are overlapping with each other (for example, banding and wrinkle). By preventing pixels from being assigned to a wrong defect type, it can improve the quality of defect severity assessment. For instance, to detect ribbon wrinkles, neighboring bright pixels in the difference image can be consecutively connected to form one or more lines. To detect printhead and/or media contamination, bright pixels located within a predetermined radius in the difference

image can be connected to form one or more voids. To detect uneven printout, such as banding or ink split, co-located bright pixels in the difference image can be assembled in groups. Secondary analysis can also be applied for further defect recognition.

(68) At step **116**, once all the bright pixels are assigned to proper defect groups, a confidence level (CL) value is assigned to each visual defect type (or refined groups), as described in Table 1 below. This value indicates how certain (“confident”) the algorithm is in detecting a particular defect type. In an embodiment, the confidence level can be calculated based on a ratio of a total amount of bright pixels in a selected area over a total amount of pixels in that area. Performing a confidence level analysis can include performing a confidence level history analysis. Additionally or alternatively, performing a confidence level analysis can include calculating confidence level values for one or more defect groups, and monitoring changes in the calculated values from one or more previous images to the current scanned image. The CL value can then be transmitted to a “history algorithm.”

(69) At step **118**, the history algorithm performs analysis for each defect type to detect how quickly the Confidence Level value evolved within one label (or from one label to another) as described in FIG. **19**. At step **120**, based on the level of the detected changes, a warning and/or error message (at step **122**) can be produced. In addition to alerting a user, at step **124**, the algorithm can also provide advice to the user on how to address the error if the system cannot resolve the issue automatically.

(70) At step **126**, the processed data can then be transmitted to update a defect database (internal or external), and used to provide an alert and/or troubleshooting method more precisely and/or more quickly in case of a future event.

(71) If, at step **120**, no defect is detected, then the method **100** may proceed with printing the next label at step **128**.

(72) FIG. **2A** shows a diagram of rescaling of a captured image to be the same size as the reference image using an example scaling algorithms. Specifically, the reference image on the leftmost is of a precise size because it is a bitmap representing the commands transmitted to the printhead. The original captured image is shown in the middle, and may represent an image obtained by an optical scanner such as a verifier. In this example, the captured image is slightly larger than the reference image (by a different factor in the horizontal and vertical directions). On the rightmost is the captured image that was rescaled to have the same dimensions as the reference image so that a pixel-by-pixel analysis can be performed.

(73) In addition to normalizing the captured image in size, the captured image's luminance, “Y,” can also be normalized to match luminance of the reference image. Luminance is a standardized value (which ranges between 0 and 1), where for a monochrome sensor 0 represents a pure black, and 1 a pure white. For a color sensor, each channel (red “R,” green “G,” and blue “B”) should be evaluated individually. For red, 0 is also a pure black, while 1 is a pure red. Similar logic is applied to the green and blue channels. The captured image is not likely to be as perfect as the reference image in terms of luminance and color. Some small deterioration may happen due to the media, heat setting, scanner element, etc., resulting in the white not being as pure white (very light grey) and the black not being as dark as it should be (dark grey color instead).

(74) A “level adjustment” can be performed in order to match the white of the captured image with white of the reference image, and the black of the captured image with black of the reference image. The formula can be applied to estimate the luminance Y for black-and-white image, as well as to the RGB channels for color image. In the formula below, Y' is the new value of the luminance, and Y is the original value. For RGB value, Y should be replaced with R, G or B based on:

$$\text{Captured}(Y') = \{\text{Captured}(Y) - \text{Captured}(\min(Y))\} * \{\text{Reference}(\max(Y)) - \text{Reference}(\min(Y))\} / \{\text{Captured}(\max(Y)) - \text{Captured}(\min(Y))\}.$$

(75) FIG. **2B** shows a result of application of the luminance equation. Specifically, representative Y' range after application of the luminance adjustment algorithm is shown. As shown by a bar **202**

on the left, the reference image reflectance ranges from 0 to 1 (i.e., 0% being black and 100% being white). The captured image will have a lower reflectance, as shown by a bar **204** in the middle. The processed captured image has been expanded to have the same reflectance as the reference image so that the processed image and reference image have the same range of luminance, as shown by a bar **206** on the right.

(76) Once the captured image has been processed to resemble the reference image in terms of size and luminance (or RGB), the difference image can be rendered. A “difference” algorithm can be used to compare the reference image and the captured image to detect a visible difference between the two. For each pixel composing the two images, a simple formula can be applied to create a difference image:

$\text{Difference\_pixel}(Y,R,G,B)=|\text{Reference\_pixel\_pixel}(Y,R,G,B)-\text{Captured\_pixel}(Y,R,G,B)|.$

(77) A “perfectly” printed label has had its size and luminance adjusted according to the present invention, and processed with the difference algorithm. A perfectly printed label will result in a perfectly black image. Any imperfections will be rendered in a grayscale range, where pure white will be a full mismatch in pixel comparison. This approach can be used to quantify printing errors beyond those simply evaluated according to a print quality standard (e.g., ISO/IEC 15415). It may be noted that the above color scheme is merely for exemplary purposes and should not be construed to limit the scope of the disclosure. Other color schemes, such as a color scheme inverse of the exemplary color scheme described above, or an entirely different color scheme may be implemented, without deviation from the scope of the disclosure.

(78) FIG. 3 shows a method **300** for generating a difference image, according to an embodiment. At step **302**, a scanned image is aligned with a reference image. At step **304**, the scanned image is scaled to match a size of the reference image. At step **306**, luminance of the scaled scanned image is adjusted. At step **308**, a pixel-to-pixel analysis is conducted between the scanned image and the reference image. And at step **310**, a difference image is generated based on results of the pixel-to-pixel analysis.

(79) In an embodiment, the method **300** can include obtaining the scanned image using a printer-verifier device. The scanned image can include an image of a printed media (e.g., a label or a tag), and may be generated by stitching together a series of consecutive linear images of the media. In an embodiment, the stitched images may be 2D in case a different imaging technology is used to acquire the images. The reference image can include a print command string, and be stored internally in the printer memory, or in an external database, such as a cloud database. The external database may be updated over time.

(80) The method **300** can also include analyzing the difference image to identify one or more printing errors. For example, the method can include detecting at least one of ribbon wrinkles, printhead and/or media contamination, platen roller contamination, black and/or white banding, and/or black and/or white ink split. Scaling the scanned image at **304** can include stretching or compressing the scanned image in a transverse direction and/or longitudinal direction.

(81) FIG. 4 shows a method **400** for image processing, according to an embodiment. At step **402**, a size of a produced image is normalized to match a size of a reference image. At step **404**, a luminance of the produced image is normalized to match a luminance of the reference image. And at step **406**, a difference image is produced by comparing the normalized produced image to the reference image.

(82) The method **400** can include capturing the produced image with an image scanning device, e.g., an integrated verifier device. Comparing the normalized produced image to the reference image at step **406** can include comparing information of the produced image to a print command string. The reference image can be stored internally, or in an external database, such as a cloud database. The database may be updated over time.

(83) In an embodiment, normalizing a luminance at step **404** can include performing level adjustment to match white and/or black colors in the produced image and white and/or black colors

in the reference image. Additionally or alternatively, normalizing a luminance at step **404** can include performing level adjustment to match red, green and/or blue colors in the produced image and red, green and/or blue colors in the reference image. Normalizing a size of a produced image at step **402** can include adjusting a size of the produced image to align corners and/or edges of the produced image with corners and/or edges of the reference image. Additionally or alternatively, normalizing a size of a produced image at step **402** can include equalizing a distance between a rightmost printed area and a leftmost printed area of the image.

(84) In an embodiment, the method **400** can include detecting one or more printing defects. For example, the printing defects can include detecting at least one of ribbon wrinkles, printhead and/or media contamination, platen roller contamination, black and/or white banding, and/or black and/or white ink split. For instance, to detect ribbon wrinkles, neighboring bright pixels in the difference image can be consecutively connected to form one or more lines. To detect printhead and/or media contamination, bright pixels located within a predetermined radius in the difference image can be connected to form one or more voids. To detect uneven printout, such as banding or ink split, co-located bright pixels in the difference image can be assembled in groups. Secondary analysis can also be applied for further defect recognition.

(85) FIG. 5 shows a method **500** for print defect detection, according to an embodiment. At step **502**, an image is scanned with a verifier to generate a captured image. At step **504**, the captured image is processed to match one or more parameters of the captured image to one or more parameters of a reference image. At step **506**, the captured image is compared to the reference image to detect one or more bright pixels. At step **508**, one or more print defects are detected.

(86) In an embodiment, processing the captured image at step **504** can include scaling, rotating, adjusting luminance and/or adjusting one or more colors. Comparing the captured image to the reference image at step **506** can include comparing a binary version of the captured image to a binary version of the reference image. Capturing an image with a verifier at step **502** can include capturing an image displaying a barcode symbol, text, and/or graphics.

(87) In an embodiment, the method **500** can include performing a bright pixel analysis. For example, performing a bright pixel analysis can include grouping and/or connecting bright pixels located within a predetermined distance from each other. For instance, neighboring bright pixels in the difference image can be consecutively connected to form one or more lines and/or voids. Additionally or alternatively, co-located bright pixels in the difference image can be assembled in groups. Secondary bright pixel analysis can be performed for further printing defect analysis.

### III. Thermal Printhead Contamination Detection

(88) In accordance with an embodiment, there are two primary indications of printhead contamination. The first is localized reduced heat conduction causing less ribbon color to be transferred for thermal transfer ribbon (TTR) printing or a lighter color for direct thermal (DT) printing (referred to as Case I herein). The second is when contamination causes the media to lose contact with the printhead and no localized color is printed (referred to as Case II herein).

(89) When contamination slowly accumulates on the printhead (Case I), the heat transfer gradually becomes less efficient. At the early stage of Case I, this is difficult to detect in the printing and is not problematic. As contamination gets worse, the printing suffers specific distortions that can be detected with a verifier. This can be illustrated with barcodes printed in “ladder” configuration.

FIG. 6A demonstrates a typical case where a barcode is printed using a contaminated thermal printhead. Specifically, the reduced heating of the ribbon, or of the thermally sensitive coating for TD printing, causes less color to be apparent on the label. Every bar is affected in this illustration, but in general the narrow bars are affected slightly more than the wide bars. Note that only the first few bars of the symbol are shown.

(90) This type of printing distortion may not cause a noticeable drop in print quality grades (e.g., as measured according to widely accepted norms ISO/IEC 15415 for 2D barcodes and ISO/IEC 15416 for linear barcodes, often called “ANSI grading”). Because grading occurs along a number of scan

paths that are perpendicular to the bars, even if one scan path hits the distortion, the other paths may average it out. Similarly, scanning performance is not adversely affected. Significant problems with this type of distortion may occur when barcode symbols are arranged in the “picket fence” configuration (e.g. where the bars are arranged perpendicular to the movement of the print media), as well as with text and/or graphics. As contamination lingers on the printhead, permanent printhead damage becomes more likely.

(91) One algorithm for detecting this type of contamination can start with identifying a barcode symbol in a verifier image and analyzing “scan lines” parallel to the motion of the media (i.e., perpendicular to the bars in the example shown in FIG. 6A). The “scan lines” are the reflectance values of every pixel in the image scanner as the media moves under the scan head. In this example, the printer-verifier includes a line scan head arranged perpendicular to the media motion that captures successive line images and assembles them into a two-dimensional image that may be nearly indistinguishable from a single image taken by a digital camera.

(92) The algorithm can compute element widths of each scan line using a suitable threshold, e.g.,  $(R_{\max} + R_{\min})/2$ . Although the image shown in FIG. 6A is “perfect,” the image from an actual printed label may show minor variations in the widths of the bars. One way to minimize this effect is to average the widths for a group of bars, e.g., all the narrow bars. Generally, there may be more than 20 or so narrow bars in a typical barcode, thus making the averaging step particularly powerful. Although FIG. 6A shows a linear (1D) barcode to exemplify the present invention, a 2D barcode (either matrix or stacked) can be analyzed as well to determine if repetitive distortion in the direction of printing is present.

(93) Once the element widths of the narrow bars are averaged for every scan line in the image, they can be plotted. It is important to note that the printing distortion from a contaminated printhead is parallel to the media motion. Consequently, distortions that do not occur in the same place in every sequentially printed area are not caused by printhead contamination. FIG. 6B shows a plot of the average narrow bar widths across the barcode symbol section shown in FIG. 6A. Here, the fact that printing distortion is located in the same place in every printed bar makes it particularly easy to detect using the averaging technique.

(94) In other embodiments, narrow spaces may be used instead of narrow bars. Additionally, linear barcodes with more than two element widths can be analyzed. Most combination of small and large bars and spaces may be combined to provide a stable measurement of Case I and Case II contamination.

(95) Another algorithm that can be used for detecting Case I distortions involves a pattern match between reference and what the verifier detects in the image. Such algorithm does not depend on the type of printing.

(96) In an embodiment, the reference may include a print command string, which is computer code that can be shown graphically (where a “to print” command is depicted as black and a “no-print” command is depicted as white). They are rearranged into rows and columns to mimic approximately what the printed image will look like. The graphically rearranged print command string can be referred to as a “reference” or “reference image.”

(97) FIG. 6C shows an example based on the image from FIG. 6A, although this technique may be equally applicable to situations where printhead contamination shows up in text, 2D barcodes, and/or graphics. Specifically, the rightmost image in FIG. 6C is an image buffer, which may be stored in the printer and eventually shifted out to the printhead when printing. The leftmost image is an illustrative image simulating a barcode printed with a contaminated printhead similar to FIG. 6A. The middle image is a bit-by-bit difference image, where a clear pattern is visible along the center-left of the barcode.

(98) Case II contamination can occur in dusty environments and/or with inexpensive media, where a speck of material adheres to the printhead, causing a complete void in the printing. Case II contamination printing looks identical to label printed with a “burned out dot” (i.e., a dot that can

no longer produce heat) in the printhead, and manifests itself as an unprinted line running from one end of the label to the other perpendicular to the printhead. This type of printhead contamination can be particularly damaging to the printhead. FIG. 6D shows an example of a printed label made with a Case II contaminated printhead.

(99) Case II contamination can be detected using similar algorithms to Case I contamination. When an integrated printer-verifier is used, the printer may be configured to detect when a dot is “burned out” by measuring printhead resistance. Therefore, if the printer shows that no dots are “burned out,” and algorithms for detecting printing caused by burned out dots return a positive value, then the printer-verifier can conclude with a high degree of certainty that there is Case II printhead contamination.

(100) Other Case II scenarios that can be detected by the present invention include but are not limited to: overheated media (some popular products on the market today may consistently fracture the thermal layer if overheated); preprinted or floodcoated direct thermal media (the inks will typically build up on the printhead more quickly than the base material); adhesives (aggressive, heavy coat weight, or cold-temperature adhesives that are prone to flow and build up on the printhead can be detected long before they become Case II contamination problems).

(101) There are various actions that the printer can be configured to take when printhead contamination is detected. For low levels of Case I contamination, for instance, the printer can transmit a message (e.g., over Wi-Fi, network, or by other means of communication, such as cellular wireless communication, sounds, and/or flashing lights) to the IT department (or other departments responsible for maintenance, such as quality control or inventory) notifying that the printer should be cleaned at the next media roll change. For higher levels of Case I contamination (e.g., when the average narrow bar widths change by more than 5%), a message can be transmitted to the maintenance department to perform a printhead cleaning process within a predetermined timeframe, such as the next hour for instance. Additionally or alternatively, a specially formatted printed label can notify the user to stop the print run and clean the printhead. For high levels of Case I contamination or when Case II contamination is detected, the printer can be configured to notify the IT department, print a label telling the user to clean the printhead immediately and/or stop the printer until the printhead is cleaned.

(102) Additionally or alternatively, the algorithm may include modifying the thermal management (firing the dot earlier, longer, or hotter) to offset the effects of Case I contamination, while notifying the user/maintenance/IT about the need to clean the printhead. This may improve print quality prior to printhead cleaning, which may be particularly valuable for printing high ANSI-grade-mandated symbols in healthcare, retail (compliance labeling for major retailers), electronics (wireless phone distribution to major carriers), and other fields relying on high-quality symbols. Degraded-quality symbols may lead to penalties in the form of returned shipments or charge-backs for increased handling (required when the pallet label does not scan, and the retailer/carrier must manually scan each carton on the pallet).

(103) Additionally, instead of an integrated printer-verifier, a stand-alone verifier (for example, as described below in connection with FIG. 22) can be used with the algorithms of the present invention. For instance, during a normal statistical quality check (e.g., every 100th label is brought to a verifier to measure its ANSI grade), the verifier can have a special setting to notify the user about printhead contamination and optionally prescribe a recommended course of action (such as to clean the printhead).

(104) The algorithms described herein can be implemented into a hand-held barcode scanner or scanner-enabled mobile terminal. In these cases, the scanner may be programmed to transmit the data to its host and perform a secondary transmission to the quality department, and/or to maintain a running file of printhead contamination for later use by the maintenance department at the end of a shift.

(105) The resulting printer's capability to validate that the dot is firing normally can enable the

printer to positively identify contamination versus a failed printhead, thereby saving the cost of an unnecessary replacement due to misdiagnosis. Additionally, the present invention can assist in defining appropriate cleaning protocols. The most common cleaning methodology is cleaning the printhead with a cloth, nonwoven card, or swab that is saturated with isopropyl alcohol. This combination may be rather inefficient for removing some common forms of buildup, such as Case II contamination, that may be better eliminated with very fine abrasive sheets. However, use of abrasive sheets on printheads without contamination will abrade the protective coatings on the printhead, leading these to be rarely recommended for fear of accelerating printhead wear. Positive identification of the type of buildup, and validating the effectiveness of cleaning methodologies, can enable a data-driven and optimized recommendation for printhead cleaning and maintenance, extending printhead life and reducing total cost of ownership.

(106) FIG. 7 shows a method **700** for determining an origin of printing distortion, according to an embodiment. At step **702**, an image of a printed barcode symbol is generated with a printer-verifier. At step **704**, the image of the printed barcode symbol is processed to detect deviation of one or more parameters of one or more elements of the barcode symbol from a predetermined threshold. At **706**, checking for a malfunction of one or more heating elements is performed. At step **708**, an origin of printing distortion is determined.

(107) In an embodiment, processing the image of the printed barcode symbol at **704** can include calculating widths of the one or more elements of the barcode symbol. The method **700** can further include averaging the calculated widths for a group of the elements of the barcode symbol. Additionally, the averaged widths can be plotted. Processing the image at **704** can also include comparing the image with a reference image. Additionally, processing the image can include processing the image with an image processor.

(108) Checking for a malfunction of one or more heating elements can include checking for a burnout of one or more heating elements. For example, the method **700** can include sensing temperature and/or resistance of the thermal printhead.

(109) FIG. 8 shows a method **800** for detecting a contaminated thermal printhead, according to an embodiment. At step **802**, a barcode symbol is identified in a verifier image. At step **804**, one or more scan lines of the identified barcode symbol from the verifier image are analyzed. At step **806**, element widths of the one or more scan lines are calculated using a predetermined threshold. At step **808**, the calculated element widths are analyzed to detect contamination of a thermal printhead.

(110) In an embodiment, identifying a barcode symbol in a verifier image at step **802** can include identifying a barcode symbol in an image obtained from a verifier device, for example an integrated printer-verifier device. Analyzing the calculated element widths at step **808** can include plotting an average deviation within a group of the element widths. Calculating element widths at step **806** can include calculating widths of one or more narrow bars and/or narrow spaces between bars of the barcode. The method **800** can further include issuing a notification reporting the detected contamination, and prescribing a recommended course of action.

(111) FIG. 9 shows a method **900** for detecting printhead contamination, according to an embodiment. At step **902**, one or more indicia are detected in an image of a printed image. At step **904**, one or more elements of the detected indicia are analyzed. At step **906**, a pattern match analysis is conducted between the detected indicia and a reference image to produce a difference image. At step **908**, the difference image is analyzed to detect a presence of printhead contamination.

(112) In an embodiment, detecting one or more indicia at step **902** can include detecting indicia using a verifier integrated into a thermal printer. Detecting indicia can include detecting one or more 1D barcodes, 2D barcodes, graphics, and/or text.

(113) The method **900** can further include reporting the presence of printhead contamination. The reporting can include sending a message to an entity responsible for printhead maintenance and/or

printing a specifically formatted label. The method **900** can further include triggering one or more predetermined actions in response to the detected contamination, wherein the predetermined actions are selected based on user sensitivity or preference to print quality. For example, some users may want to know only if there is a speck of contamination that impacts scannability, while others may be highly sensitive to minor fluctuations in print quality.

#### IV. Platen Roller Contamination and Wrinkle Defects Detection

(114) FIG. **1000** shows an image processing method **1000**, according to an embodiment. At step **1002**, an image of a label is captured. At step **1004**, the captured image is processed to produce a difference image having a plurality of bright pixels by comparing the captured image to a reference image. At step **1006**, the bright pixels located within a predetermined radius are consecutively connected to form a line until there are no pixels within the radius left to connect. At step **1008**, the pixels are iteratively connected until all the pixels of the plurality of bright pixels having neighboring pixels within the predetermined radius are connected, and one or more lines are formed.

(115) In an embodiment, consecutively connecting the pixels at step **1006** can further include monitoring a running average slope defining an orientation of the line being formed. Monitoring a running average slope can include determining an angle of the slope with a point-slope technique. Additionally, the method **1000** can include making a connection between two consecutive pixels when a resulting change in the running average slope does not exceed a predetermined angle threshold (such as a predetermined static and/or dynamic angle value). The method **1000** can further include monitoring an average direction of the running average slope, and making a connection between two consecutive pixels when such connection follows a forward direction of the slope.

##### (116) A. Wrinkle Detection

(117) FIG. **11** shows a method **1100** for determining ribbon wrinkle, according to an embodiment. At step **1102**, a difference image is created to locate one or more bright points by comparing a captured image of a media after printing to a reference image. At step **1104**, the bright points located near each other are grouped to form one or more primary lines characterized by a running average slope. At step **1106**, the primary lines having a similar running average slope are connected to form one or more secondary lines.

(118) In an embodiment, the method **1100** can include assigning a confidence level value to the one or more primary and/or secondary lines. Additionally, the method **1100** can include requesting capturing an additional image to replace the captured image having one or more primary and/or secondary lines with low confidence level values.

(119) Creating a difference image at step **1102** can include using the reference image stored in a self-learning database. Depending on an embodiment, the database can be external or internal. Additionally, comparing a captured image of a media can include comparing a captured image of a media displaying a barcode.

(120) FIG. **12A** shows a ribbon wrinkle detection method **1200**, according to an embodiment. At step **1202**, a barcode symbol having a plurality of elements displayed on a media is identified. At step **1204**, the barcode symbol is surrounded with a bounding box encompassing top and bottom parts and outer edges of the barcode symbol, and/or one or more finder patterns. At step **1206**, one or more unprinted points located near the elements of the barcode symbol are located. At step **1208**, co-localized unprinted points are connected to form one or more lines. At step **1210**, an angle of the one or more lines relative to the bounding box is determined. At step **1212**, verification that each of the determined angles exceeds a predetermined threshold value is made.

(121) In an embodiment, identifying a barcode symbol at step **1202** can include identifying a two-dimensional barcode symbol. For example, FIG. **12B** graphically depicts a 2D barcode (left) and the 2D barcode surrounded with a bounding box **1201** (right). Alternatively, identifying a barcode symbol at step **1202** can include identifying a linear barcode symbol. For example, FIG. **12C**



graphically depicts a linear barcode (top) and the linear barcode surrounded with a bounding box **1203** (bottom). Additionally, the method **1200** can include determining a number of the one or more lines, and/or displaying a result of the angle verification. Determining an angle at step **1210** can include determining an angle with a point-slope technique. Additionally or alternatively, determining an angle at **1210** can include using a linear regression technique. Verifying that each of the determined angles exceeds a predetermined threshold value at step **1212** can include exceeding a predetermined dynamic angle value, and/or a predetermined fixed angle value.

(122) FIG. **13A** shows a label printed with a wrinkled ribbon. By comparing a computer memory image to be printed to an actual image of the label, one can determine the difference between the intended and the final printed label. As a result, everything that was the same in the intended graphic and the image of the printed label can be shown in black because of a subtractive process. Where there is a difference, any imperfections can be rendered in a grayscale range (where pure white can refer to a full mismatch in pixel comparison). FIGS. **13B** and **13C** show difference images produced for the label printed with a wrinkled ribbon.

(123) In an embodiment, the difference image can be analyzed using the following algorithm to highlight the bright pixels: based on the (x; y) coordinate of each pixel, vectors between the pixels, their respective slopes, and an average distance between the pixels can be calculated. One can then connect each pixel depending on their proximity with their neighbors as well as the slope change from pixel to pixel, and the void space existing between the pixels.

(124) In an embodiment, a connection algorithm can use the following pixel structure to analyze and group the pixels together during the analysis. Each pixel, noted "P," can have three states: 1. P.sub.free: the pixel has not been utilized in the algorithm yet; 2. P.sub.used: the pixel has been used and connected to another pixel; and 3. P.sub.bad: the pixel is a random dot with no near, usable neighbor pixel.

(125) In addition, each pixel can be assigned a group (1, 2, 3 . . . ) representing the number of an individual wrinkle line found. Hence, a pixel can be defined in a structure with the following information: P{x; y; state; group}. The starting pixel P can be chosen based on P(min(x);min(y)). Its group can then be assigned to 1 and its state can be set to P.sub.used.

(126) To connect the pixels based on distance, a radius variable can be defined to help determine if a bright pixel belongs to the same group. The algorithm can first check if there is any pixel within the predetermined radius to connect to. If none are present, then the pixel can be marked as P.sub.bad, meaning that it is a random defect and that it should be ignored. For a pixel that has a neighbor within the vicinity, the algorithm can then check if there are still any pixels free. In this case, P can be connected to the nearest free pixel P'. Once the connection is done, P' can then be assigned the same group number as P, and its state can be changed to P.sub.used; P' can become the new starting point, and the algorithm can then try to connect P' to the next nearest pixel to form a line. In the case that all nearest pixels are used, the algorithm can jump to:

$P''(\max(x);\max(y)) \in P.\text{sub.Used}(P(\text{group})),$   
and P'' can become a new starting point.

(127) In the case where P'' has no more free neighbors (all marked P.sub.used within the predetermined radius), the algorithm can mark the end of a line for the current wrinkle. The algorithm can then move to the next P.sub.free pixel, increasing the group number by one to detect a new wrinkle line. The algorithm can continue until there are no more free pixels; then the algorithm can end, as all or almost all wrinkles have been detected.

(128) In an embodiment, when attempting to link a pixel, a ribbon wrinkle detection algorithm can also take into consideration a slope change when looking at the nearest pixel. The algorithm can calculate a running average slope to determine an orientation of the wrinkle being currently analyzed. If the slope change exceeds a predetermined threshold (+/-60 degrees, for example), then the connection may not be made unless the distance between P and P':

$\text{dist}(P,P') < (\text{Radius}/k),$

where  $k$  is a rational and positive number.

(129) If the slope does not carry information on the direction of the line, the algorithm may also check if the connection to  $P'$  is going “backward” from the existing average direction. Again, the connection may not be made in this case unless  $\text{dist}(P, P') < (\text{Radius}/p)$  is a rational and positive number.

(130) In some cases, some wrinkle lines have a similar slope, and are running on the similar average line (even though they appear disconnected at first because the edge of the wrinkle detected may be separated by more than the radius distance). The algorithm may connect such lines together to make them belong to the same group. FIG. 13D shows three different wrinkle lines detected separately during a first pass of the algorithm, shown as lines **1302**, **1304**, and **1306**, respectively. All the lines are located on the similar running average slope, meaning that they can belong to the same group. FIG. 13E shows reconnection of the three lines **1302**, **1304**, and **1306** from FIG. 13D as belonging to the same wrinkle group (shown on the left side of the figure as a single gray line **1308**) for a final analysis and detection. FIG. 13C shows the type of connections made and a resulting pattern. Specifically, FIG. 13C shows a pattern made by connecting the pixels in a difference image of a label that was printed with a wrinkled ribbon and/or media. These white lines in the difference image can be analyzed and recorded in a verification report. Optionally, the printer can display a message or send a message to the host suggesting that the user adjust the ribbon and/or media.

(131) In an embodiment, the algorithm can also perform a defect analysis, by having the printer either access a database of defects stored internally in the printer memory, or access an external database, such as a cloud database. The external database may be updated to include a history of printing issues that can be found over time. During the defect analysis, the algorithm may also assign a confidence level (CL) value indicating how certain it is in detecting the current issue, for example a wrinkle. A high CL value can be given when the defect detected matches one or more criteria in the wrinkle database. In case of lower CL value, one or more additional printed samples may be required before triggering an error message. Additionally, the database can be self-learning based on the number of occurrence and frequency of the ribbon wrinkle issue in order to define the root cause and best solution more accurately over time.

(132) B. Void/Platen Roller Contamination Detection

(133) FIG. 14A shows a label printed with a platen roller contamination. FIG. 14B shows difference images produced for the label printed with a contaminated platen roller. FIG. 14C shows a relationship between the bright points of FIG. 14B.

(134) By comparing a computer memory image to be printed to an actual image of the label, one can determine the difference between the intended and the final printed label. As a result, everything that was the same in the intended graphic and the image of the printed label can be shown in black because of a subtractive process. Where there is a difference, any imperfections can be rendered in a grayscale range where pure white can refer to a full mismatch in pixel comparison.

(135) In an embodiment, an algorithm can be created to determine whether the images in FIGS. 14B and 14C were caused by platen roller contamination based on the bright points. A bright point caused by platen roller contamination may be defined as one or more areas of white whose total extent falls within a circular area (for example, an area of no more than 0.120 inches (36 pixels in a 300 dpi difference image)). If there is more than one bright point, then each other bright point has to be separated by at least a predetermined distance (for example 0.15 inches (or 45 pixels)). If there are three to six bright points, then no three bright points can lie on a straight line. If there are more than six bright points, then no group of five bright points can lie on a straight line. If these criteria are met, then the printed label can be deemed to have defects caused by a contaminated platen roller. After evaluation of the white points, a platen roller contamination remedy process can be executed.

(136) In an embodiment, to highlight the bright pixels, the difference image may be analyzed using

the following algorithm. Because each pixel has its own (x; y) coordinate, vectors between the pixels, their respective slopes, and an average distance between the pixels can be calculated. One can then connect each pixel (or group each cluster of pixels, in case of analyzing printing defects caused by dust, grit, etc., here referred to as “void analysis”) depending on their proximity with their neighbors as well as the slope change from pixel to pixel, and the void space existing between the pixels.

(137) In an embodiment, a connection algorithm can use the following pixel structure to analyze and group the pixels together during the analysis. Each pixel, noted “P,” can have three states: 1. P.sub.free: the pixel has not been utilized in the algorithm yet; 2. P.sub.used: the pixel has been used and connected to another pixel; 3. P.sub.bad: the pixel is a random dot with no near, usable neighbor pixel.

(138) In addition, each pixel can be assigned a group (1, 2, 3 . . . ) representing the number of an individual void found. Hence a pixel can be defined in a structure with the following information: P{x; y; state; group}. The starting pixel P can be chosen based on P(min(x);min(y)). Its group can then be assigned to 1 and its state can be set to P.sub.used.

(139) To connect the pixels based on distance, a radius variable can be defined to help determine if a bright pixel belongs to the same group. The algorithm can first check if there is any pixel within the predetermined radius to connect to. If none are present then the pixel can be marked as P.sub.bad, it is a random defect and should be ignored. For a pixel that has a neighbor within the vicinity, the algorithm can then check if there are still any pixels free. In this case, P can be connected to the nearest free pixel P'. Once the connection is done, P' can then be assigned the same group number as P, and its state can be changed to P.sub.used. P' can become the new starting point, and the algorithm can then try to connect P' to the next nearest pixel to form a cluster of pixels marking the void. In the case that all nearest pixels are used, the algorithm can jump to:  $P''(\max(x);\max(y)) \in P.sub.used(P(group))$ , and P'' can become a new starting point.

(140) In the case where P'' has no more free neighbors (all marked P.sub.used within the predetermined radius), the algorithm can mark the end of a cluster for the current Void group. The algorithm can then move to the next P.sub.free pixel, increasing the group number by one to detect a new void mark. The algorithm can continue until there are no more free pixels; then the algorithm can end, as all or almost all voids have been detected. FIG. 14D highlights four different void groups made using the above algorithm (left) and a zoom on one of the voids (right), showing all the pixels belonging to the same void group.

(141) In an embodiment, the algorithm can also perform a defect analysis by having the printer either access a database of defects stored internally in the printer memory, or access an external database, such as a cloud database. The external database may be updated to include a history of printing issues that can be found over time. During the defect analysis, the algorithm may also assign a confidence level (CL) value indicating how certain it is in detecting the current issue, for example a void. A high CL value can be given when the defect detected matches with different criteria of void detection when the comparison is performed with void database. In case of lower CL value, one or more additional printed samples may be required before triggering an error message. Additionally, the database can be self-learning based on the number of occurrence and frequency of the voiding issue, in order to define the root cause and best solution more accurately over time.

(142) In an embodiment, an image processing method can include recognizing a void pattern. A void pattern can include following unique particularities: void points can be small, making the total ratio of bright spots in comparison black surface small; the groups of points indicating a void can be separated by distances much greater than the distances between individual void points. If any of these criteria are fulfilled, the method can ensure that the voids are detected.

(143) Each group can be delimited by its own lowest pixel and highest pixel in the X, Y direction.

Thus, a boundary can be made using  $P(\min(x);\min(y))$  and  $P(\max(x);\max(y))$  with both pixels belonging to the same Group  $j$ . Simple rectangular or circular boundaries can be formed.

(144) In one embodiment, a method of checking for voids can include calculating the “ratio of coverage,” where the surface covered by the void over the label surface can be calculated, followed by checking if it is lower than a predefined ratio “VoidCoefficient” to detect a void:

$\text{TotalVoidSurface/LabelSurface} < \text{VoidCoefficient}$

When this comparison is true, a void is detected.

(145) In another embodiment, a method of checking for voids can include using the distance between each group, as the distance is likely to be rather large compared to a pixel/dot size. The algorithm can calculate the average distance between each group and compare it to a defined value “VoidDistance”:

$\text{AverageDistanceBetweenGroups} > \text{VoidDistance}$

If the comparison is true, then a void is detected.

(146) In an embodiment, the method can include determining if the void is caused by contamination on the platen roller or media. To determine if one or more void marks are due to contamination on the platen roller or the media, the algorithm can check for a repeat pattern. The platen roller has a fixed circumference, which means that the majority of the void marks caused by platen roller contamination will repeat at a fixed distance approximately equal to the platen roller circumference.

(147) For a long label, the repeat void group can be on the same X-axis while separated by a fixed Y distance equal to the platen roller circumference, plus or minus a small margin for measurement error. FIG. 14E shows an example of repeating void marks due to platen roller contamination.

(148) In an embodiment, once it is determined that a platen roller or the media is contaminated, a printer-verifier can communicate to the user or to the host system the nature and extent of the contamination. For instance, if two or three bright points are found, the printer can issue a low priority message to clean the platen roller at the next media change. If the label similar to the one shown in FIG. 14A is produced (which has at least six void areas), the printer can issue a “Stop Print” notification so the operator or maintenance team can address the problem immediately. Alternatively, a fully automated method can be installed in the printer such as one or more nozzles connected to a compressed air supply. When platen roller contamination is detected, a brief blast of compressed air in close proximity to the roller can be initiated. The following label can then be analyzed according to the method described herein to see if the voiding is reduced or eliminated. The process can be repeated, if necessary.

## V. Multiple Defects Detection

(149) When more than one defect occurs within one label, refining the grouping is especially important. For example, a label printed with a combination of a void and banding defects is shown in FIG. 15. FIG. 16 shows a label printed with wrinkle and banding issues. Left-hand sides of FIGS. 15 and 16 display images of the labels, whereas right-hand sides show difference images obtained by comparing scanned images of those labels to corresponding reference images. In FIG. 16, areas of the defect overlap are marked with boxes in the difference image. In such areas, bright pixels can belong to both types of defect, and thus further refinement may be necessary.

(150) Once the difference image is produced, one or more defect detection algorithms can be applied to detect printing defects of various types (such as ribbon wrinkle, banding, ink split, etc.) by producing rendered images, where bright pixels are assigned to corresponding defect groups. Such rendered images can then be overlapped and analyzed to determine if any bright pixels are common to more than one output image. When such pixels are found in two or more images, additional image processing methods can be used to refine, extract and reassign each pixel to its correct defect group.

(151) FIGS. 17A-17D show an example of a label containing wrinkle and banding defects before and after being refined by the algorithm. Specifically, FIG. 17A shows ribbon wrinkle analysis

output when both wrinkle and banding issues are present. In FIG. 17A, outlined with the boxes are areas of incorrect detection before refining the group. FIG. 17B shows banding analysis output when both wrinkle and banding issue are present. In FIG. 17B, outlined with the boxes are areas of incorrect detection before refining the group. FIG. 17C shows refined wrinkle analysis with bright pixels belonging to the banding defect removed from the image. FIG. 17D shows refined banding analysis with bright pixels belonging to the wrinkle defect removed from the image.

(152) Each refined group is assigned a Confidence Level value. The Confidence Level (CL) value aims to indicate how certain the system is in detecting an error. Using the refined data, the algorithm can calculate such values as a ratio of bright pixels over the total number of pixels of the entire scanned image as well as within each group. Based on the ratio value for each defect type, the algorithm can either use a look-up table to retrieve the Confidence Level value, or it can calculate it by using one or more formulas. A simplified flow is described below for an ease of understanding; more components can be taken into account to improve the confidence level calculation.

(153) For example, each group found can contain at least the following information obtained from a corresponding refined image output: the x-coordinate of the leftmost pixel defined by Group.n(min(x)); the x-coordinate of the rightmost pixel defined by Group.n(max(x)); the y-coordinate of the lower-most pixel defined by Group.n(min(y)); the y-coordinate of the upper-most pixel defined by Group.n(max(y)); the number of bright pixels within the group defined as Group.n(#bright pixels); and the surface area covered by the group in number of pixels, defined by: Group.nTotalPix=Group.n(max(x)–min(x))×(max(y)–min(y)).

(154) Using this information, the algorithm can calculate the ratio of bright pixels over total number of pixels in each group. For example, FIG. 18 shows a group information data used for calculating such ratio. The area is delimited by the top right and bottom left pixels, and is outlined with a box in the figure. The equation to determine the ratio:

$$(155) \text{Group} . n_{W\%ratio} = \frac{\text{Group} . n(\#brightpixels)}{\text{Group} . nTotalPix} .$$

(156) In addition to the ratio within a group, the algorithm can calculate the ratio of bright pixels over total number of pixels for the entire image:

$$(157) \text{Image}_{W\%ratio} = \frac{\text{Image}(\#brightpixels)}{\text{Image}(\#pixels)} ,$$

where Image(#bright pixels)=Σ.sub.i=1.sup.kGroup.i(#bright pixels), with k being the number of group found in the picture. Additionally, further definition is possible:

Image(#pixels)=image.PixHeight×image.PixLength.

(158) Several methods can be used to assign the Confidence Level values in the process of detecting visual print quality defects and identifying which defect is observed. For example, such methods can use the “image.sub.W % ratio” and/or the “Group.n.sub.W % ratio” to compare to predefined values, which may be stored in a look-up table. Table 1 shows a simplified example of the look-up table. The values listed in Table 1 are provided for illustrative purposes only.

(159) TABLE-US-00001 TABLE 1 Confidence level values for various printing defect types.

Image W %	Confidence Level	value ratio, %	Void	Wrinkle	Ink Split	Banding
0-3	60	1	1	0	3-5	85
14	5	0	5-15	15	67	17
0	15-25	3	19	65	1	25-30
0	3	74	4	30-45	0	0
27	19	45-70	0	0	0	73
70-90	0	0	0	0	0	0
54	90-100	0	0	0	12	

(160) For example, if the ratio found is 17% based on the refined image for the void detection, then the CL value is 3, indicating a low chance that void is present. However, if the ratio is 1%, the CL value jumps to 60, showing an above average probability that void type contamination is present in the label.

(161) Alternatively or additionally, assigning confidence level value to various defect types can involve other calculations. For example, the following equation can be used:

$$(162) CL = A \times e^{-\frac{1}{2} \times \frac{(\text{imageBWratio} - \text{offset})^2}{\text{spread}^2}} ,$$

where values A, offset and spread are unique to each defect type, and can have either fixed or

dynamic values.

(163) In an embodiment, the algorithm can be configured to determine what type of action, warning or error message to trigger in order to alert the user and solve the issue, based on the current and/or past CL data.

(164) In cases when the detected CL value is high, an error message may be triggered immediately. However, some printing defects may be progressive in nature, and exhibit a stronger effect over time. For example, when a platen roller is wearing out, at first the effect in creating banding is minimal, and it can be difficult to determine whether it is a true issue or not. To overcome this kind of uncertainty, the history algorithm can be configured to monitor how the Confidence Level value varies over time, and determine whether to activate an error correction sequence.

(165) FIG. 19 shows an exemplary embodiment of the history algorithm 1900. Such a flow chart can be applied to each error type. For illustrative purposes, variables in the depicted flow chart do not identify the error type, although each variable can be unique to the error type, e.g., CL.n can refer to CL.n.void, CL.n.wrinkle, etc.

(166) At step 1902, the history algorithm 1900 collects CL.n values for each defect type. At step 1904, the history algorithm 1900 calculates the CL Slope, which shows how fast the CL values changes from previous image to current image. At step 1906, the history algorithm 1900 determines whether the CL Slope exceeds a certain threshold.

(167) When the CL Slope does not exceed the threshold at step 1906, the algorithm 1900 can then check if there is a trend of a negative CL slope. For example, the algorithm 1900 may calculate the total value of negative slope as shown at step 1908, and determine whether the total value of negative slope exceeds a threshold (noIssueCnt) at step 1910. If this is the case, the Total CL value for the current defect can be reset to 0 at step 1912, and the next sample is collected at step 1914. If no, there is no reset of the Total CL value.

(168) When the CL slope exceeds the defined threshold at step 1906, the algorithm 1900 can then verify the slope value at step 1916. Based on the result of verification, the algorithm 1900 can increase the CL.Total value at steps 1918 and 1920, and may additionally increasing the total value of positive slope observed at step 1918.

(169) At step 1922, the CL.Total value can then be checked to verify if the Total Confidence Level has reached an error threshold to trigger the alarm and/or initiate corrective actions. If the CL.Total value exceeds the error threshold, then the algorithm 1900 determines whether error type makes printout unreadable at step 1928. If so, then the algorithm 1900 triggers error warning at step 1926. If error type does not make printout unreadable, the algorithm 1900 further determines whether the total value of positive slope exceeds a threshold (RapidErrorCnt) at step 1930. If so, then the algorithm 1900 may trigger an error warning at step 1926. If not, the algorithm 1900 may proceed with collecting the next sample at step 1914.

(170) At step 1922, if the CL.Total value is increasing slowly and steadily without reaching the error threshold, the algorithm 1900 examine whether the total value of positive slope exceeds a safety limit at step 1924. If so, the algorithm 1900 can notify the user and/or mark the issue as highly probable to occur on a future label to be printed at step 1926. If not, the algorithm 1900 may proceed with collecting the next sample at step 1914.

(171) Through the history algorithm, the system can be configured to trigger a corrective or preventative action (such as alert or error message) when a certain visual defect type is detected. There are various actions that the printing system can be configured to take when a printing defect is detected. For instance, the printer can transmit a message over Wi-Fi, network, or by other means of communication, such as cellular wireless communication, sounds, and/or flashing lights to the IT department or other departments responsible for maintenance, such as quality control or inventory. A message can also be transmitted to the maintenance department to perform a maintenance process within a predetermined timeframe, such as the next hour for instance. Based on the type of the defect, the printer can also provide guidance on how to resolve the issue. Additionally or

alternatively, a specially formatted printed label can notify the user to stop the print run and/or take one or more predetermined actions to address the detected issue. The algorithm may also include modifying the thermal management setting, while notifying the user/maintenance/IT.

(172) Additionally, error detection feedback can be provided to an internal and/or external database, and/or an external storage unit such as a cloud database. The external cloud database can be updated based not only on local data of onset printer, but also include history of visual printing defect issues that can be found over time and over a wide range of printers at various locations. The database can be self-learning, and take into account the frequency of occurrence of each detected issue, to improve determination of the root cause and/or issue resolution in the future.

(173) Using previously obtained data, the database can be configured to dynamically update various algorithms, such as the CL value assignment and/or calculation. For example, if, for a certain defect, the Confidence Level value always (or often) slowly increases, and it takes about 50 labels to reach the threshold, then after detecting this trend enough times, certain actions can be taken during its next occurrence, such as lowering the threshold or boosting up the confidence level value assigned to it, so that it takes only 15 labels to reach the trigger level in the future.

Additionally, the algorithm can be configured to provide an improved output rendered image to facilitate future defect detection.

(174) FIG. 20 shows a method **2000** for printing defect analysis, according to an embodiment. At step **2002**, an image of a printout on a media is captured. At step **2004**, the captured image is checked for one or more printing defects. At step **2006**, evolution of the detected printing defects between the current captured image and one or more images of preceding printouts is analyzed. At step **2008**, results of the evolution analysis are used to determine if one or more predetermined corrective actions should be initiated.

(175) In an embodiment, checking for printing defects at step **2004** can include generating and analyzing a difference image obtained by comparing the captured image to a reference image, and/or processing the captured image to detect printhead, platen roller and/or media contamination. Generating a difference image can include comparing a binary version of the captured image to a binary version of the reference image. Checking the captured image for one or more printing defects can include detecting at least one of ribbon wrinkles, printhead, platen roller and/or media contamination, black and/or white banding, and/or black and/or white ink split.

(176) In an embodiment, analyzing evolution at step **2006** can include comparing one or more evolution characteristics (such as confidence level values) of the defects to a predetermined threshold. The method **2000** can further include verifying the one or more evolution characteristics when one or more characteristics are found to exceed the predetermined threshold, and determining if one or more predetermined corrective actions should be initiated. Additionally or alternatively, the method **2000** can further include initiating one or more predetermined corrective actions (such as preemptive maintenance) when the one or more evolution characteristics are slowly rising without exceeding the predetermined threshold. Initiating corrective actions can include (but are not limited to) triggering an alert, producing an error message, stopping printer operation, and/or prescribing a recommended course of action.

(177) Additionally, the method **2000** can include providing feedback to a self-learning defect database. The provided feedback can then be used to dynamically update one or more algorithms for checking the image for printing defects, and/or for analyzing evolution of the detected defects. Capturing an image of a printout on a media at step **2002** can include scanning an image with a printer-verifier device.

## VI. Additional Implementation Details

(178) In the specification and figures, typical embodiments of the invention have been disclosed. The present invention is not limited to such exemplary embodiments. The use of the term “and/or” includes any and all combinations of one or more of the associated listed items. The figures are schematic representations and so are not necessarily drawn to scale. Unless otherwise noted,

specific terms have been used in a generic and descriptive sense and not for purposes of limitation. (179) Device and method components are meant to show only those specific details that are pertinent to understanding the embodiments of the present disclosure so as not to obscure the disclosure with details that will be readily apparent to those of ordinary skill in the art having the benefit of the description herein. In various embodiments, the sequence in which the elements of appear in exemplary embodiments disclosed herein may vary. Two or more method steps may be performed simultaneously or in a different order than the sequence in which the elements appear in the exemplary embodiments, unless indicated otherwise.

(180) Various embodiments of the present invention may be implemented in a printer connected to a verifier, or a printer-verifier. The processor of the printer or the verifier may carry out steps of methods in accordance with various embodiments of the present invention.

(181) Referring now to FIGS. **21A-21B**, an exemplary printer-verifier **2100** (printing mechanism) capable of printing on print media **2112** is partially shown. The depicted printer-verifier **2100** of FIG. **21A** has a body **2118** for enclosing an interior thereof. The printer-verifier **2100** further comprises a power source and a moveable cover for accessing the interior and any components therein.

(182) In various embodiments, the printer-verifier **2100** is a thermal transfer printer-verifier that includes a ribbon supply spindle **2130** contained within the body **2118**. A ribbon supply roll **2108** is configured to be disposed on the ribbon supply spindle **2130**. The ribbon supply roll **2108** comprises ink ribbon **2102** wound on a ribbon supply spool **2104**. The ink ribbon supplies the media (e.g., ink) that transfers onto the print media. The printer-verifier **2100** may further comprise a thermal printhead **2116** utilized to thermally transfer a portion of ink from the ink ribbon **2102** to the print media **2112** as the ink ribbon is unwound from the ribbon supply spool **2104** along a ribbon path (arrow B in FIG. **21A**), and the print media **2112** is unwound from a media supply spool **2114** along a media path (arrow C in FIG. **21A**).

(183) A media supply roll **2110** comprises the print media **2112** wound on the media supply spool **2114**. A media supply spindle **2132** on which the media supply roll **2110** is configured to be disposed is contained within the body **2118**. A ribbon rewind spindle **2134** on which unwound ribbon is wound up may also be contained within the body **2118**. A ribbon take-up **2106** may be disposed on the ribbon rewind spindle **2134**, although the ribbon take-up **2106** on the ribbon rewind spindle **2134** may not be necessary.

(184) The printer-verifier **2100** may further comprise one or more motors for rotating the ribbon supply spindle **2130** and the ribbon supply roll **2108** disposed thereon (if present) in a forward (arrow A in FIG. **21A**) or a backward rotational direction (dependent on the ink surface), for rotating the media supply roll **2110** disposed on the media supply spindle **2132** in a forward rotational direction, and for rotating the ribbon rewind spindle **2134**. In a thermal direct printer-verifier, the ribbon supply spool, the ribbon rewind spool, and the ribbon may be eliminated and a thermally sensitive paper places the print media. These components are also included in a printer-verifier **2100** as described above.

(185) The printer-verifier **2100** may include a GUI **2122** for communication between a user and the printer-verifier **2100**. The GUI **2122** may be communicatively coupled to the other components of the printer-verifier for displaying visual and/or auditory information and receiving information from the user (e.g., typed, touched, spoken, etc.). As depicted in FIG. **21A**, the body **2118** of the printer-verifier **2100** may include the GUI **2122** with, for example, a display **2124** and a keypad **2126** with function buttons **2128** that may be configured to perform various typical printing functions (e.g., cancel print job, advance print media, and the like) or be programmable for the execution of macros containing preset printing parameters for a particular type of print media. The graphical user interface (GUI) **2122** may be supplemented or replaced by other forms of data entry or printer control, such as a separate data entry and control module linked wirelessly or by a data cable operationally coupled to a computer, a router, or the like. The GUI **2122** may be



operationally/communicatively coupled to a processor (CPU) **2120** for controlling the operation of the printer-verifier **2100**, in addition to other functions. In some embodiments, the user interface may be other than depicted in FIG. **21A**. In some embodiments, there may not be a user interface. (186) Referring now to FIG. **21B**, an example block diagram of the printer-verifier **2100** is shown. The printer-verifier **2100** may comprise the processor **2120**, a memory **2140** communicatively coupled to the processor **2120**, and a power source. The printer may further comprise a communications module **2142** communicatively coupled to one or more of the other printer components.

(187) The central processing unit (CPU) (i.e., the processor **2120**) is the electronic circuitry within a computer that carries out the instructions of a computer program by performing the basic arithmetic, logical, control and input/output (I/O) operations specified by the instructions as described above. The printer-verifier **2100** may be communicatively connected using the communications module **2142** to a computer or a network **2144** via a wired or wireless data link. In a wireless configuration, the communications module **2142** may communicate with a host device over the network **2144** via a variety of communication protocols (e.g., WI-FI®, BLUETOOTH®, CDMA, TDMA, or GSM). In accordance with various embodiments of the present invention, the memory **2140** is configured to store a print quality verification program **2146**, a reference image **2148**, an offset value **2150**, and a drifting offset value **2152** as described above.

(188) Still referring to FIGS. **21A** and **21B**, an imaging module **2136** is disposed in the printer-verifier **2100** and is configured to capture a representation of the printed image (e.g., printed barcode **2154** on print medium **2112** within a field of view **2156**), using an image sensor **2158** (i.e., the imaging module **2136** comprises the image sensor **2158**) to obtain a captured image. The image sensor **2158** comprises a light source **2160** for illuminating the field of view. The image sensor **2158** uses an imaging lens (or lenses) to form a real image of the field of view **2156** on an array of photo sensors (e.g., a linear or 2D array CCD, CMOS sensor, etc.). Electronic signals from the photo sensors are used to create gray level or color images, which would result in a digital image similar to that which may be obtained by a digital camera. The processor **2120** is further configured to carry out steps of methods as described above in accordance with various embodiments of the present invention.

(189) Referring now to FIG. **22**, an example printer **2228** (printing mechanism) communicatively coupled to verifier **2202** in system **2200** for printing an image and verifying a print quality of the image is shown. Printer **2228** may be similar to the printer-verifier **2100** depicted in FIGS. **21A-21B**, except that the imaging module of the verifier is separated from the printer in system **2200**.

(190) Similar to the printer-verifier **2100** described above in connection with FIGS. **21A-21B**, the printer **2228** may comprise a processor, a memory communicatively coupled to the processor, and a power source. The printer may further comprise a communications module communicatively coupled to one or more of the other printer components. The printer **2228** may have a fewer or greater number of components as described above.

(191) The verifier **2202** comprises imaging module **2236**, a memory (a verifier memory **2214**) communicatively coupled to the imaging module **2236** and a central processing unit (CPU) (herein a “verifier processor” **2210**) communicatively coupled to the verifier memory **2214** and imaging module **2236**. The verifier **2202** may further comprise an I/O module **2222** and a verifier communication module **2216**.

(192) The subsystems in the verifier **2202** of FIG. **22** are electrically connected via a coupler (e.g., wires, traces, etc.) to form an interconnection subsystem. The interconnection system may include power buses or lines, data buses, instruction buses, address buses, etc., that allow operation of the modules/subsystems and the interaction there between. The I/O module **2222** may include a verifier graphical user interface. In various embodiments, the verifier **2202** may be communicatively connected using the verifier communication module **2216** to the computer or the network **2218** via a wired or wireless data link. In a wireless configuration for the wireless data link, the verifier

communication module **2216** may communicate with a host device, such as the computer, or the network **2218**, via a variety of communication protocols (e.g., WI-FI®, BLUETOOTH®, NFC®, RFID®), CDMA, TDMA, or GSM). The verifier memory **2214** may store a print quality verification program **2220**, the reference image **2223**, the offset **2224**, and the drifting offset **2226**. (193) While FIG. **22** depicts a verifier memory **2214** and a verifier processor **2210** in the verifier **2202**, it is to be understood that only the printer **2228** or only the verifier **2202**, or both the printer **2228** and verifier **2202** communicatively coupled thereto may comprise the memory and the processor for executing the steps as described above (i.e., at least one of the verifier and the printer comprises a memory communicatively coupled to the imaging module and a processor communicatively coupled to the imaging module and memory). The verifier **2202** that is attached to the printer may rely on the memory and the processor of printer for executing the steps as described above while the verifier **2202** that is a standalone device has its own verifier memory **2214** and verifier processor **2210** for executing the steps as described above. Additionally, or alternatively, the printer may rely on the verifier memory **2214** and the verifier processor **2210** of verifier **2202** attached to the printer for executing the steps as described above.

(194) The imaging module **2236** disposed in verifier **2202** is configured to capture the representation of the printed image (e.g., the printed barcode **2201** on the print media **2212** in FIG. **22**) within a field of view **2203**, using the image sensor **2204** (i.e., the imaging module **2236** comprises the image sensor **2204**). The image sensor **2204** comprises the light source **2206** for illuminating the field of view. The image sensor **2204** uses an imaging lens (or lenses) to form a real image of the field of view **2203** on an array of photo sensors (e.g., a linear or 2D array CCD, CMOS sensor, CIS device, etc.). Electronic signals from the photo sensors are used to create gray level or color images, e.g., which would result in a digital image that may be obtained by a digital camera.

(195) While a thermal transfer printer-verifier and printer are described, it is to be understood that various embodiments of the present invention may be used in other types of printers (e.g., ink-drop printer, laser-toner printer, etc.).

## VII. Incorporation By Reference

(196) To supplement the present disclosure, this application incorporates entirely by reference the following commonly assigned patents, patent application publications, and patent applications:

U.S. Pat. Nos. 6,832,725; 7,128,266; 7,159,783; 7,413,127; 7,726,575; 8,294,969; 8,317,105; 8,322,622; 8,366,005; 8,371,507; 8,376,233; 8,381,979; 8,390,909; 8,408,464; 8,408,468; 8,408,469; 8,424,768; 8,448,863; 8,457,013; 8,459,557; 8,469,272; 8,474,712; 8,479,992; 8,490,877; 8,517,271; 8,523,076; 8,528,818; 8,544,737; 8,548,242; 8,548,420; 8,550,335; 8,550,354; 8,550,357; 8,556,174; 8,556,176; 8,556,177; 8,559,767; 8,599,957; 8,561,895; 8,561,903; 8,561,905; 8,565,107; 8,571,307; 8,579,200; 8,583,924; 8,584,945; 8,587,595; 8,587,697; 8,588,869; 8,590,789; 8,596,539; 8,596,542; 8,596,543; 8,599,271; 8,599,957; 8,600,158; 8,600,167; 8,602,309; 8,608,053; 8,608,071; 8,611,309; 8,615,487; 8,616,454; 8,621,123; 8,622,303; 8,628,013; 8,628,015; 8,628,016; 8,629,926; 8,630,491; 8,635,309; 8,636,200; 8,636,212; 8,636,215; 8,636,224; 8,638,806; 8,640,958; 8,640,960; 8,643,717; 8,646,692; 8,646,694; 8,657,200; 8,659,397; 8,668,149; 8,678,285; 8,678,286; 8,682,077; 8,687,282; 8,692,927; 8,695,880; 8,698,949; 8,717,494; 8,717,494; 8,720,783; 8,723,804; 8,723,904; 8,727,223; 8,740,082; 8,740,085; 8,746,563; 8,750,445; 8,752,766; 8,756,059; 8,757,495; 8,760,563; 8,763,909; 8,777,108; 8,777,109; 8,779,898; 8,781,520; 8,783,573; 8,789,757; 8,789,758; 8,789,759; 8,794,520; 8,794,522; 8,794,525; 8,794,526; 8,798,367; 8,807,431; 8,807,432; 8,820,630; 8,822,848; 8,824,692; 8,824,696; 8,842,849; 8,844,822; 8,844,823; 8,849,019; 8,851,383; 8,854,633; 8,866,963; 8,868,421; 8,868,519; 8,868,802; 8,868,803; 8,870,074; 8,879,639; 8,880,426; 8,881,983; 8,881,987; 8,903,172; 8,908,995; 8,910,870; 8,910,875; 8,914,290; 8,914,788; 8,915,439; 8,915,444; 8,916,789; 8,918,250; 8,918,564; 8,925,818; 8,939,374; 8,942,480; 8,944,313; 8,944,327; 8,944,332; 8,950,678;

8,967,468; 8,971,346; 8,976,030; 8,976,368; 8,978,981; 8,978,983; 8,978,984; 8,985,456;  
8,985,457; 8,985,459; 8,985,461; 8,988,578; 8,988,590; 8,991,704; 8,996,194; 8,996,384;  
9,002,641; 9,007,368; 9,010,641; 9,015,513; 9,016,576; 9,022,288; 9,030,964; 9,033,240;  
9,033,242; 9,036,054; 9,037,344; 9,038,911; 9,038,915; 9,047,098; 9,047,359; 9,047,420;  
9,047,525; 9,047,531; 9,053,055; 9,053,378; 9,053,380; 9,058,526; 9,064,165; 9,064,165;  
9,064,167; 9,064,168; 9,064,254; 9,066,032; 9,070,032; 9,076,459; 9,079,423; 9,080,856;  
9,082,023; 9,082,031; 9,084,032; 9,087,250; 9,092,681; 9,092,682; 9,092,683; 9,093,141;  
9,098,763; 9,104,929; 9,104,934; 9,107,484; 9,111,159; 9,111,166; 9,135,483; 9,137,009;  
9,141,839; 9,147,096; 9,148,474; 9,158,000; 9,158,340; 9,158,953; 9,159,059; 9,165,174;  
9,171,543; 9,183,425; 9,189,669; 9,195,844; 9,202,458; 9,208,366; 9,208,367; 9,219,836;  
9,224,024; 9,224,027; 9,230,140; 9,235,553; 9,239,950; 9,245,492; 9,248,640; 9,250,652;  
9,250,712; 9,251,411; 9,258,033; 9,262,633; 9,262,660; 9,262,662; 9,269,036; 9,270,782;  
9,274,812; 9,275,388; 9,277,668; 9,280,693; 9,286,496; 9,298,964; 9,301,427; 9,313,377;  
9,317,037; 9,319,548; 9,342,723; 9,361,882; 9,365,381; 9,373,018; 9,375,945; 9,378,403;  
9,383,848; 9,384,374; 9,390,304; 9,390,596; 9,411,386; 9,412,242; 9,418,269; 9,418,270;  
9,465,967; 9,423,318; 9,424,454; 9,436,860; 9,443,123; 9,443,222; 9,454,689; 9,464,885;  
9,465,967; 9,478,983; 9,481,186; 9,487,113; 9,488,986; 9,489,782; 9,490,540; 9,491,729;  
9,497,092; 9,507,974; 9,519,814; 9,521,331; 9,530,038; 9,572,901; 9,558,386; 9,606,581;  
9,646,189; 9,646,191; 9,652,648; 9,652,653; 9,656,487; 9,659,198; 9,680,282; 9,697,401;  
9,701,140; U.S. Design Pat. Nos. D702,237; D716,285; D723,560; D730,357; D730,901;  
D730,902; D734,339; D737,321; D754,205; D754,206; D757,009; D760,719; D762,604;  
D766,244; D777,166; D771,631; D783,601; D785,617; D785,636; D790,505; D790,546;  
International Publication No. 2013/163789; U.S. Patent Application Publication No.  
2008/0185432; U.S. Patent Application Publication No. 2009/0134221; U.S. Patent Application  
Publication No. 2010/0177080; U.S. Patent Application Publication No. 2010/0177076; U.S. Patent  
Application Publication No. 2010/0177707; U.S. Patent Application Publication No.  
2010/0177749; U.S. Patent Application Publication No. 2010/0265880; U.S. Patent Application  
Publication No. 2011/0202554; U.S. Patent Application Publication No. 2012/0111946; U.S. Patent  
Application Publication No. 2012/0168511; U.S. Patent Application Publication No.  
2012/0168512; U.S. Patent Application Publication No. 2012/0193423; U.S. Patent Application  
Publication No. 2012/0194692; U.S. Patent Application Publication No. 2012/0203647; U.S. Patent  
Application Publication No. 2012/0223141; U.S. Patent Application Publication No.  
2012/0228382; U.S. Patent Application Publication No. 2012/0248188; U.S. Patent Application  
Publication No. 2013/0043312; U.S. Patent Application Publication No. 2013/0082104; U.S. Patent  
Application Publication No. 2013/0175341; U.S. Patent Application Publication No.  
2013/0175343; U.S. Patent Application Publication No. 2013/0257744; U.S. Patent Application  
Publication No. 2013/0257759; U.S. Patent Application Publication No. 2013/0270346; U.S. Patent  
Application Publication No. 2013/0292475; U.S. Patent Application Publication No.  
2013/0292477; U.S. Patent Application Publication No. 2013/0293539; U.S. Patent Application  
Publication No. 2013/0293540; U.S. Patent Application Publication No. 2013/0306728; U.S. Patent  
Application Publication No. 2013/0306731; U.S. Patent Application Publication No.  
2013/0307964; U.S. Patent Application Publication No. 2013/0308625; U.S. Patent Application  
Publication No. 2013/0313324; U.S. Patent Application Publication No. 2013/0332996; U.S. Patent  
Application Publication No. 2014/0001267; U.S. Patent Application Publication No.  
2014/0025584; U.S. Patent Application Publication No. 2014/0034734; U.S. Patent Application  
Publication No. 2014/0036848; U.S. Patent Application Publication No. 2014/0039693; U.S. Patent  
Application Publication No. 2014/0049120; U.S. Patent Application Publication No.  
2014/0049635; U.S. Patent Application Publication No. 2014/0061306; U.S. Patent Application  
Publication No. 2014/0063289; U.S. Patent Application Publication No. 2014/0066136; U.S. Patent  
Application Publication No. 2014/0067692; U.S. Patent Application Publication No.

[illegible]

[illegible]

[illegible]

2017/0193432; U.S. Patent Application Publication No. 2017/0193461; U.S. Patent Application Publication No. 2017/0193727; U.S. Patent Application Publication No. 2017/0199266; U.S. Patent Application Publication No. 2017/0200108; and U.S. Patent Application Publication No. 2017/0200275.

## Claims

1. A method comprising: capturing an image of printed content; generating a difference image based on a comparison of the captured image of the printed content and a reference image, wherein the reference image correlates to the printed content; and generating a plurality of rendered images based on applying a plurality of defect detection algorithms to the difference image, wherein each of the plurality of rendered images comprises one or more bright pixels that are assigned to at least one defect group from a plurality of defect groups.
2. The method of claim 1, wherein the plurality of defect detection algorithms comprises one or more of a void detection algorithm, a wrinkle detection algorithm, or a banding and ink split detection algorithm.
3. The method of claim 1, wherein each of the plurality of defect groups is associated with one of a plurality of defect types comprising void, wrinkle, banding and ink split.
4. The method of claim 3, further comprising determining at least one defect type associated with the printed content from the plurality of defect types.
5. The method of claim 1 further comprising overlapping and analyzing the plurality of rendered images to determine whether at least one of the one or more bright pixels is common among two or more of the plurality of rendered images.
6. The method of claim 5 further comprising, in response to determine that the at least one of the one or more bright pixels is common among the two or more of the plurality of rendered images, performing additional processing of the plurality of rendered images.
7. The method of claim 6, wherein performing the additional processing of the plurality of rendered images comprises overlapping the at least one of the one or more bright pixels that is common.
8. The method of claim 7, wherein performing the additional processing of the plurality of rendered images comprises refining the one or more bright pixels assigned to the plurality of defect groups.
9. The method of claim 7, wherein refining the one or more bright pixels assigned to the plurality of defect groups comprises reassigning one of the one or more bright pixels from one defect group to another defect group of the plurality of defect groups.
10. A printing system comprising: a printer; and a verifier communicatively coupled to the printer and comprising: a processing unit; a memory storing computer program code, the memory and the computer program code configured to, with the processing unit, cause the verifier to at least: identify one or more groups of difference pixels; refine the one or more identified groups of difference pixels into one or more defect types; determine a confidence level for at least one of the one or more defect types; and determine one or more defects based on the determined confidence level for at least one of the one or more defect types.
11. The printing system of claim 10, wherein the memory and the computer program code are configured to, with the processing unit, cause the verifier to further create a group of difference points in response to determining that a difference point is within a predetermined distance of another difference point.
12. The printing system of claim 11, wherein the memory and the computer program code are configured to, with the processing unit, cause the verifier to further detect a printhead contamination based on a count of difference pixels in respective groups of difference pixels.
13. The printing system of claim 11, wherein the memory and the computer program code are configured to, with the processing unit, cause the verifier to further determine whether the group of difference points is collinear based on attempting to connect three difference points in the group of

difference points on a straight line.

14. The printing system of claim 13, wherein the memory and the computer program code are configured to, with the processing unit, cause the verifier to further determine that the one or more defects are caused by wrinkles on a ribbon in response to determining that at least one or more difference points are collinear.

15. The printing system of claim 13, wherein the memory and the computer program code are configured to, with the processing unit, cause the verifier to further determine that the one or more defects are caused by a contaminated platen roller in response to determining that at least one or more difference points are non-collinear.

16. The printing system of claim 10, wherein the memory and the computer program code are configured to, with the processing unit, cause the verifier to further refine the one or more identified groups of difference pixels by creating a separate difference image for each detected defect type of void, ribbon wrinkle, banding, and ink split.

17. The printing system of claim 10, wherein the printer is configured to issue a notification reporting a defect.

18. The printing system of claim 10, wherein the printer is configured to prescribe a recommended course of action corresponding to at least one or more identified defect types.

19. The printing system of claim 10, wherein the one or more groups of difference pixels are identified using a reference image that is generated based on a print command string.

20. The printing system of claim 19, wherein the one or more groups of difference pixels are identified by comparing the reference image with a printed image.

---